

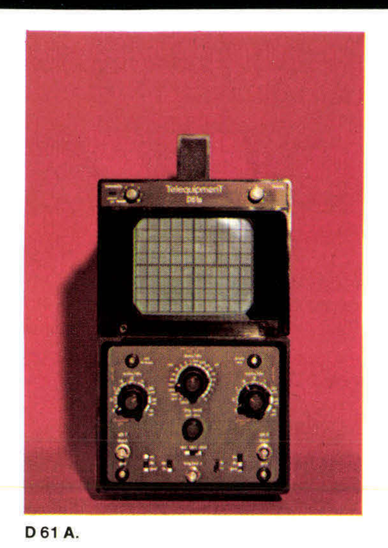

MICROPROCESSEURS / MICRO-ORDINATEURS / INFORMATIQUE APPLIQUÉE

N° 4 Bimestriel – Mars/Avril 1979

**10F**

Suisse 4,50 F - Belgique 81 F.B - Canada 2,50 $C - Italie 2.100 Lires - Espagne 150 Ptas - Algérie 10 Din. - Tunisie 1.000 Mil.

# Chez PENTASONIC 6 mois de crédit gratuit sur les oscilloscopes.

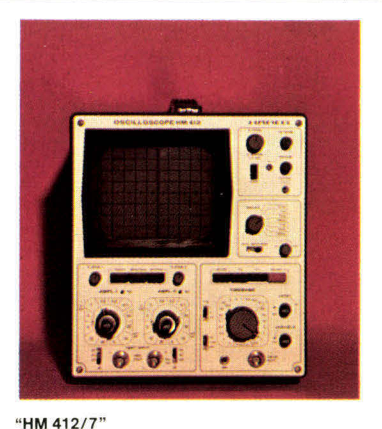


D 61 A.

D 65.



D 67 A.



"HM 412/7"

D 1010 -

## TELEQUIPMENT

**1700 F** — **S 61 - 5 MHz**
Dimensions 28 x 16 x 37 cm. Tube 8 x 10 cm. Grande luminosité. Ampli vertical. bande pass. : 0 à 5 MHz.

**2820 F** — **D 61 A. Double trace 10 MHz**
Surface utile de l'écran : 8 x 10 cm. Bande passante : 10 MHz à 10 mV/cm.

**4854 F** — **D 65. Double trace 15 MHz**
Surface utile de l'écran : 8 x 10 cm. Bande passante : 15 MHz à 10 mV/cm.

**7680 F** — **D 67 A. Double trace 2 x 25 MHz**
1 mV/cm à 50 V/cm. Double base de temps. Balayage retardé. Déclenchement : normal, AC, DC, TV ligne et trame, automatique, HF-REJ.

### NOUVELLE GAMME "D 1000"

**2920 F** — **D 1010 - Double trace 10 MHz**
5 mV à 20 V/div. Tension maxi 500 V. Balayage 0,2 s à 0,2µS/div., 40 ns en X5.

**3231 F** — **D 1011 - double trace 10 MHz**
version plus performante du 1010 mais caractéristiques principales identiques.

**3880 F** — **D 1015 - Double trace 10 MHz**
5 mV - 20 V/div. Tension maxi 500 V - Balayage 0,2 s à 0,2µS/div. 40 ns en X5.

**4464 F** — **D 1016 - Double trace 15 MHz**
version plus performante du 1015 mais caractéristiques principales identiques.

*Les 4 appareils de cette nouvelle série sont conçus à partir d'une technologie avancée, garantie d'une grande simplicité d'emploi. Ils sont particulièrement adaptés à l'enseignement, au dépannage TV, à la fabrication et au contrôle industriel.*

## HAMEG

**1445 F** — **"HM 307"**
Simple trace. DC - 10 MHz (– 3 dB). Entrée à 12 possibilités ± 5 %. 5 mVcc - 20 Vcc/div.

**2446 F** — **"HM 312" Double trace 2 x 10 MHz**
Sensibilité 5 mV/cm à 20 V/cm. Déclenchement LPS - Tube 8 x 10 cm.

**3269 F** — **"HM 412/7" Double trace 2 x 15 MHz**
Tube 8 x 10 cm. AMPLIFICATEUR VERTICAL. Bande passante DC : à 15 MHz (– 3 dB), à 20 MHz (– 6 dB). Sensib. : 5 mVcc - 20 Vcc/cm. Balayage retardé.

**5045 F** — **"HM 512/7" Nouveau double trace 2 x 40 MHz**
2 canaux DC à 40 MHz, ligne à retard. Sensib. : 5 mVcc - 20 Vcc/cm. Régl. fin 1:3. Dim. de l'écran : 8 x 10 cm. Graticule lumineux.

**SONDES OSCILLOSCOPES**
Commutation X1 - X10 sur la sonde. Prix...... **192 F**

## LEADER

**3580 F** — **"LBO 508". Double trace 2 x 20 MHz.**
10 mV/cm. Soustraction de trace XY. Base de temps 0,5µs à 200 mS/cm.

## VOC

**4234 F** — **"VOC 5". Double trace 15 MHz**
Bandes passantes : DC de 0 à 15 MHz (– 3 dB), AC de 2 à 15 MHz (– 3 dB).

★★★★★★★★★

**L'expédition de nos appareils n'est pas gratuite, mais :**

- *Ils voyagent aux risques et périls de PENTASONIC.*
- *Ils ne sont pas expédiés par la poste, ni par la S.N.C.F., mais par un transporteur.*
- *Ils sont assurés. Si jamais un de nos appareils présente à l'arrivée (vérifiez avec le transporteur) le moindre défaut d'aspect, il vous sera immédiatement changé à nos frais.*

**EMBALLAGE - TRANSPORT - ASSURANCE**
En contre-remboursement, **78 F** -
Avec chèque à la commande, **53 F.**

★★★★★★★★★

**CREDIT IMMEDIAT**

☎ **524-23-16**
**SUR LE PONT DE GRENELLE**
**5, rue Maurice-Bourdet - 75016 PARIS**
A 50 mètres de la Maison de la Radio
Autobus : **70-72** (arrêt MAISON DE L'ORTF). MÉTRO : **Charles-Michels**

☎ **331-56-46**
**AUX GOBELINS**
**10 boulevard Arago - 75013 PARIS**
*Métro : Gobelins*

*Ouvert tous les jours (sauf dimanche) de 9 h à 12 h 30 et 14 h à 19 h 30*

# MICRO SYSTEMES

S

15, rue de la Paix - 75002 Paris - Tél. : 296.46.97.

Illustration François COLLET

# SOYEZ




présent au
Salon des Composants
du 2 au 7 avril 1979
(Zone "Presse")
et à
Micro-Expo
du 15 au 17 mai 1979

# PRÉSENT DANS LE FUTUR

**Comme nous, vous appartenez à un monde où tout est en perpétuel mouvement. L'électronique et l'informatique n'échappent pas à la règle.**

**Dans cet univers technologique où tout vieillit vite et contre lequel nous ne pouvons rien, il est important de savoir profiter du présent pour être capable d'affirmer sa présence dans le futur.**

**MICRO-SYSTÈMES peut certainement vous y aider.**

**A quoi sert le progrès s'il ne profite pas à tout le monde.**

S'il est superflu de dire que l'usage du microprocesseur et du micro-ordinateur se développe rapidement en France, tant chacun devient quotidiennement le témoin de cet irréversible raz-de-marée technologique, force est de reconnaître que ce phénomène ne constituera un réel progrès que dans la mesure où il sera offert à tous la possibilité d'aborder avec un minimum de connaissances cette activité naissante qu'est l'électronique programmée.

MICRO-SYSTÈMES est né en réponse à ce besoin.

**Carrefour entre l'électronique et l'informatique.**

Première revue française d'initiation et de développement à la micro-informatique, MICRO-SYSTÈMES constitue un carrefour entre l'électronique et l'informatique, destiné à réunir l'ensemble des professionnels, étudiants et amateurs qui souhaitent participer à l'essor de cette technique au champ d'applications à peine exploré.

**Dans MICRO-SYSTÈMES ce sont des spécialistes qui écrivent.**

Paraissant tous les deux mois, MICRO-SYSTÈMES présente pour chaque numéro un ensemble de noms prestigieux du monde de la micro-informatique dont la collaboration éclairée apporte au travers de leurs rubriques initiation, réalisations, programmation, une solution à tous ceux qui veulent s'initier ou se perfectionner à la logique programmée, soit à des fins professionnelles, soit en tant que loisir.

**MICRO-SYSTÈMES : un rendez-vous permanent avec la réalité.**

Face à de telles ambitions, MICRO-SYSTÈMES ne peut pas se contenter d'être une simple revue qui se laisse confortablement lire et vous abondonne ensuite à la dernière page. En plus de nos articles, il s'agit d'offrir à nos lecteurs un ensemble de services qui soit pour eux le moyen de pénétrer sur le terrain du vécu, de marquer l'actualité de leur empreinte.

C'est ainsi que le 25 novembre 1978, à Paris, 1.000 lecteurs assistaient pour la première fois à un séminaire exceptionnel MICRO-SYSTÈMES.

**Construisez votre micro-ordinateur : MICRO-SYSTÈMES 1.**

Pour parvenir à une parfaite maîtrise des microprocesseurs, il est bon de passer de la théorie à la pratique. C'est la raison pour laquelle, parmi les réalisations qu'elle proposera, la rédaction s'est fixée pour objectif fondamental de rendre ses lecteurs capables de mener à bien l'étude et la construction de leur propre micro-ordinateur : MICRO-SYSTÈMES 1. Programmable en BASIC, il est accessible aux non-spécialistes de l'informatique.

**MICRO-SYSTÈMES : une porte ouverte sur un monde nouveau.**

MICRO-SYSTÈMES ne s'attache pas à savoir si ses lecteurs sont des professionnels ou des amateurs. Ce qui est certain c'est qu'ensemble ils préparent dès maintenant l'avenir et que pour eux cette revue peut être la clé de la micro-informatique ; une porte ouverte sur un monde nouveau.



# Ne courez plus après l'information

**Sachez économiser votre temps et votre argent en recevant chez vous votre numéro de MICRO-SYSTÈMES.**

Si vous aviez été un abonné régulier, vous auriez pu suivre dans nos récents numéros, tous ces sujets :

**"Le choix d'un micro-processeur," "Initiation aux micro-processeurs," "Le Basic," "Alarme antivol temporisée à micro-processeurs," "Réalisez votre micro-ordinateur," "Les micro-ordinateurs individuels : mythe ou réalité," "Générateur de fonctions à micro-processeur," "Système de vérification des mémoires mortes," "Programme financier," "Jeux sur micro-ordinateur : le Startrek," "Quel micro-ordinateur choisir ?"**

Chacun de ces sujets aurait pu vous apporter une aide appréciable dans vos décisions professionnelles ou personnelles.

MICRO-SYSTÈMES est là pour vous conseiller et vous informer sur tout ce que la micro-informatique peut constituer de nouveau pour vous.

Ne manquez plus votre rendez-vous avec MICRO-SYSTÈMES. Abonnez-vous dès maintenant et profitez de cette réduction qui vous est offerte.




# Bulletin d'abonnement

## 1 an - 6 numéros - 45 F

**Bimestriel (France 45 F - Etranger 70 F)**

(1) ☐ Je m'abonne pour la première fois à partir du numéro paraissant au mois de :

☐ Je renouvelle mon abonnement et je joins ma dernière étiquette d'envoi.

Je joins à ce bulletin la somme de : ______ F par :
- chèque postal ☐ sans numéro de compte
- chèque bancaire ☐
- mandat-lettre ☐

à l'ordre de MICRO-SYSTÈMES.

*(1) ☐ Mettre une croix dans les cases ci-dessus correspondantes.*

**Informations :**

**Pour les changements d'adresse :** joindre la dernière étiquette d'envoi, ou à défaut, l'ancienne adresse accompagnée de la somme de 2,40 F en timbres-poste, et des références complètes de la nouvelle adresse.

**Pour tous renseignements** ou réclamations, concernant votre abonnement, joindre la dernière étiquette d'envoi.

*Ecrire en capitales, n'inscrire qu'une lettre par case. Laisser une case entre 2 mots. Merci.*

|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

*Nom, prénom (attention : prière d'indiquer en premier lieu le nom suivi du prénom).*

|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

*Complément d'adresse (résidence, chez M., bâtiment, escalier, etc.).*

|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

*N° et rue ou lieu-dit*

|_|_|_|_|_|   |_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|_|

*Code postal* *Ville*

| Dept | Cne | Quartier |
|---|---|---|
| | | |

13 19

*Ne rien inscrire dans ces cases.*

Bulletin à retourner à MICRO-SYSTÈMES
Service des abonnements
2 à 12, rue de Bellevue,
75940 Paris Cedex 19. Tél. 200.33.05.

# Sommaire

Pages






MICRO SYSTEMES







Photo André Edouard.

***Notre couverture :***

*« Choix d'un microprocesseur » (p. 52) : le microprocesseur rapide 4 bits SF.C 92901 et son micro-séquenceur.*

*Un hexagone déformé par un micro-ordinateur dans un rapport donné génère une courbe appelée* ***hexapuce.*** *Cette figure a été réalisée par l'A.D.A.O. de Dijon (p. 39).*


Président-directeur général :
Directeur de la publication :
**Jean-Pierre Ventillard**

Rédacteur en chef :
**Alain Tailliar**

**Conseiller technique :** Dave Habert

**Comité de rédaction :**
Jean-Michel Cour — André Doris
Jean Frémaux — Dave Habert
Alain Tailliar — Jean-José Wanègue

**Secrétaire :**
Catherine Salbreux

**Rédaction :**
15, rue de la Paix, 75002 Paris
**Tél. : 296.46.97**

**Maquette :** Josiane Garnier.

**Publicité :**
S.P.E. — **Chef de publicité :** M. Sabbagh
**Tél. : 607.34.58**

**Abonnements :**
2 à 12, rue de Bellevue
75940 Paris Cedex 19
**Tél. : 200.33.05**
1 an (6 numéros) : 45 F (France)
70 F (Etranger)

**Société Parisienne d'Edition**
Société anonyme au capital de 1 950 000 F
Siège social : 43, rue de Dunkerque, 75010 Paris
Direction - Administration - Ventes :
2 à 12, rue de Bellevue, 75940 Paris Cedex 19
Tél. : 200.33.05 - Télex PGV 230472 F





Dépôt légal 3e trimestre 78
N° éditeur : 644
Distribué par SAEM Transports Presse
Imprimerie : La Haye-Mureaux


Micro-Systèmes décline toute responsabilité quant aux opinions formulées dans les articles. Celles-ci n'engageant que leurs auteurs.

# PENTA-MAGAZINE

**TOUS RENSEIGNEMENTS MICROPROCESSEURS**
**RIVE DROITE 524.23.16**
**RIVE GAUCHE 331.56.46**
**OU SUR PLACE**

*Editorial*

## Oui ! la concurrence, ça stimule !!!

Outre de nouveaux accords de prix avec SESCOSEM, PENTASONIC importe directement ses composants de son bureau de NEW YORK !

### MAINTENANT EN FRANCE, LES PRIX AMÉRICAINS

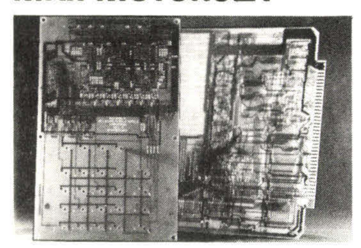
TOUS LES COMPOSANTS POUR LA CARTE MICROSYSTÈME I SONT DISPONIBLES

Mémoire SV 6726 1 pièce . . . **184,00 F** 8 pièces . . . **151,00 F** 16 pièces . . . **123,00 F**
Mémoire GC 3 1 pièce . . . **194,00 F** Mémoire DC 3 . . . **35,70 F**
ROM BASIC . . . **980,00 F**

### IMPORTÉS DIRECTEMENT D'ALLEMAGNE

2103 - 1024 Bits 450 nS (à partir d'une pièce) **12,00 F**
4116 - RAM 16 K Dyn., 350 nS . . . . . . . . . . . . . . **87,00 F**
2114 L 45 - RAM 1024 x 4. 450 nS . . . . . . . . . . . **72,00 F**

## Le "NUMÉRO 1" ou l'anti-gadget : MKII MOTOROLA



Le fait de spécifier qu'il est à la base du 6800 MOTOROLA suffit à le considérer comme l'un des systèmes les plus souples, les plus performants, mais surtout, l'un des plus faciles à utiliser.
Il existe des microprocesseurs plus puissants mais, à notre avis, aucun de plus sympathique. Allié à sa facilité d'emploi (manuel d'utilisation en français), vous bénéficiez d'un service après-vente digne de MOTOROLA.
La vocation du MK II, outre l'initiation, est d'être le premier maillon d'un système puissant capable de gérer télétype, visu et floppy; d'être extensible grâce à ses cartes interfaces... qui existent!
Le MK II c'est l'anti-gadget des systèmes de développement.
*1720 F avec notice d'utilisation en français.*

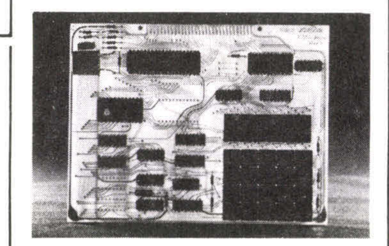
## L'OUTSIDER : le VIM I SYNERTEK

Nous considérons le 6502 comme un microprocesseur très légèrement moins puissant que le 6800, bien que très proche de celui-ci. Le système de développement qui lui est associé est, par contre, très étoffé. Ses 4 K de moniteur d'origine, son 1 K de RAM, sa gestion directe de télétype ou de visu, en font un système complet. La plupart des extensions peuvent se monter sur la carte et particulièrement 3 K de RAM supplémentaires, ainsi que 32 K de ROM. Parmi ces 32 K de ROM on pourra implanter un gros BASIC de 8 K, dont Synertek annonce la commercialisation. Autre avantage le VIM I se vend tout monté.
*2280 F avec notice d'utilisation en français.*

## FAITES-LE DONC VOUS-MÊME!

Voulez-vous construire votre système 6800 vous-même? Nous tenons, dans ce cas, à votre disposition un classeur comprenant le circuit imprimé, les plans, et les notices d'un système de développement. Une fois monté, il gérera une visu ou un télétype. Il disposera d'1 K de mémoire, un PIA, un ACIA et un moniteur MIK-BUG. Ce kit, destiné aux amateurs avertis, bénéficie de l'assistance technique PENTASONIC. MOTOROLA le commercialise, dans sa version d'origine sous l'appellation MKI.
*Le circuit imprimé 200 F.*

## Un petit malin qui cache son jeu

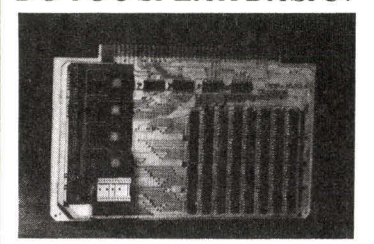


Ce produit à base de SCMP détient certainement l'un des meilleurs rapports prix/performances du marché. Particulièrement destiné aux contrôles de processus, c'est le système d'initiation par excellence. Toutes les cartes d'extension, ainsi que le produit lui-même sont fabriqués en France.
*Série UC 1000 EMR, 985 F*

## *LA MEILLEURE ET LA PIRE DES CHOSES*

Le soft est la meilleure et la pire des choses. La meilleure lorsqu'il s'agit d'aider l'utilisateur par une plus grande souplesse de langage, une plus grande puissance. La pire lorsqu'on tombe dans la facilité et qu'on l'utilise mal : par exemple gérer un clavier ou une visu directement. Le soft n'est pas le Bon Dieu! Il ne peut pas tout faire et s'il s'occupe à servir ses propres intérêts, il ne lui restera que peu de temps à vous consacrer.
Le but, notre but, n'est pas d'afficher des petits caractères sur un printer ou sur une télévision, mais, de travailler en concordance avec notre système. N'oubliez jamais que le soft n'est que le prolongement de vous-même : avec vos qualités et ses défauts. Du soft... PENTABUG ! qui se présente sous la forme de 1 MM 2716 se montant directement sur les emplacements EPROM du MK II et permet de brancher directement nos cartes de visualisation et nos claviers sur les KITS MAZEL et MK II • Mis au point par M. DAUTREVAUX..................... 195 F TTC
Comprenant 1 MM 2716 + 1 notice + 1 listing

## *DO YOU SPEAK BASIC?*



Cette carte, fabriquée par PROTEUS INTERNATIONAL vous permet de compléter votre système informatique et d'accéder à un "personal computer" de haut de gamme. Il existe différents types de Basic. Le Tiny-Basic (de 2 à 4 K), qui se rapproche plus du gadget que de l'outil de travail et qui permet entre autre d'apprendre le langage. Le Full-Basic (de 4 à 7 K) qui est le langage d'origine et l'Extended-Basic que l'on considère comme le plus souple, et l'un des plus puissants. Grâce à celui-ci, vous aurez accès à la fantastique bibliothèque existante : calcul scientifique, gestion de stock, aterrissage de fusée, jeux, etc.
*Carte basic 1820 F*

- Ce moniteur comprend et exécute 28 fonctions.
- Il permet en outre d'adresser un ACIA en 8010-8011 et de commander tout interface travaillant en ASC II.

## PET chez PENTA

**MICRO ORDINATEUR DE MARQUE COMMODORE.**

Microprocesseur 6502 (Mos Technology). Clavier 73 touches. Ecran 23 cm, 25 lignes de 40 caractères. Magnétophone à K7. Mémoire RAM 8 K dont 7 utilisables. ROM 14 K. BUS IEEE 488. Pseudo graphisme.

Import H.T. .................... **6250 F**

TTC .......................... **7350 F**

## APPLE II

- Microprocesseur Rockwell 6502 RAM extensible de 4 à 48 K.
- Basic-Monitor-Assembleur-Désassembleur (ROM).
- Sortie vidéo 24 lignes/40 colonnes.
- Graphiques fins en couleurs sur TV (R.V.B. SECAM).
- Interfaces magnétophone et entrées analogiques. Haut-parleur incorporé.
- 8 périphériques connectables dont :
  - — Imprimante, Modem, carte de communication RS 232.
  - — Carte de reconnaissance vocale (32 mots quelconques).
  - — Floppy disques (1 à 14 fois 116 Ko).

* Dos : fichiers de données en accès séquentiel indexé, programmatique/chainage des programmes/protections d'écriture, etc.

Prix H.T. ...................... **8333 F**

Prix TTC ...................... **9799 F**

Unité de disquette TTC ........ **5490 F**

Soit un rapport 44,10/kilo octet.

## PROTEUS III

**Micro-ordinateur de marque PROTEUS INTERNATIONAL**

Equipé du microprocesseur 6800.
Clavier 53 touches iapacitives.
Sortie vidéo ou VHF 16 lignes de 64 UCARACT1RES.
PSEUDO GRAPHISME.
Majuscules-Minuscules.
Vitesse d'écriture 1 800 bauds.
Interface cassette KANSAS CITY.
Dispose du BUS P III.
Mémoire RAM 17 K dont 16 utilisables.
ROM 8 K.
D'origine : sorties RS 232, TTY, TTL, VIDEO, UHF, MAGNETOPHONE.

**PRIX H.T.** .................... **7500 F**

**PRIX T.T.C.** .................. **8820 F**

**PRIX VERSION 32 K TTC** .. **10758 F**

Plutôt orienté vers la gestion des PME, commerce, professions libérales, sa puissance lui permet d'être en outre, un produit hobbyste de haut de gamme.

## FLOPPY PROTEUS

Prix pour 1 disquette .. **7820 F**

2 disquettes .. **11518 F**

3 disquettes .. **15216 F**

Ces prix incluent le DOS 16 K. Dans la configuration 3 disquettes, on dispose de 1,5 millions octets de mémoire, soit un rapport de 10 F le K octet.

**CONFIGURATION D'UN DOS**

**DOS se compose de deux parties :**

- Un ensemble de fonctions systèmes, utilisables en assembleur, permettant d'exploiter le plus efficacement et le plus facilement possible l'ensemble des ressources matérielles du système. (Floppys disques en particulier.)
- Un jeu de processeurs interactifs permettant un accès rapide depuis la console à l'ensemble de ces ressources. Ces processeurs sont :
- **BACKUP.** Permet d'effectuer des copies, des vérifications ou des réorganisations de disquettes entières (nécessite un minimum de 2 floppys).
- **CHAIN.** Permet d'enchainer l'exécution de processeurs système ou utilisateurs.
- **COPY.** Permet la recopie de fichiers.
- **DELETE.** Permet de détruire des fichiers.
- **CAT.** Permet de lire tout ou partie du catalogue des fichiers.
- **INIT.** Permet d'initialiser une nouvelle disquette. (Nécessite un minimum de 2 floppys).
- **FREE.** Permet de connaitre la place disponible sur le disque et dans le catalogue.
- **LIST.** Permet de lister un fichier.
- **LOAD.** Permet de charger un fichier en mémoire.
- **CHANGE.** Permet de changer, le nom, le suffixe, les attributs ou les clefs d'un fichier.

D'autre part, **DOS** est prévu pour faciliter la programmation dans divers langages grâce aux processeurs suivants :

- **EDIT.** Permet d'éditer le texte d'un programme d'un langage quelconque, avec des facilités accrues.
- **ASM.** Permet d'écrire et d'utiliser des programmes en langages ASSEMBLEUR.
- **RASM.** Possède toutes les possibilités de la commande ASM, plus la possibilité de définir des macro-instructions.
- **RLOAD.** Permet l'utilisation de programmes translatables, créés par RASM.
- **BASIC.** Permet l'utilisation de l'interpréteur BASIC-DOS, étendant les possibilités du BASIC résident (en lui donnant accès au disque souple, entre autres).

N.B. — Ces 5 processeurs, sauf BASIC, sont disponibles sur option.

Tous les processeurs DOS travaillent sur des fichiers qui sont identifiés comme suit, par :

- Un nom de fichier, de 1 à 12 caractères alpha-numériques.
- Un suffixe, de 1 ou 2 caractères alphabétiques.
- Une clef de lecture, de 6 caractères alphanumériques (optionnelle).
- Une clef d'écriture, de 6 caractères alphanumériques (optionnelle).

Si le fichier a été créé avec une clef de lecture, elle doit être précisée pour toute lecture du fichier.
Si le fichier a été créé avec une clef d'écriture, elle doit être précisée pour toute écriture sur le fichier.
Pour détruire le fichier ces deux clefs doivent être précisées.
Par ailleurs, toujours pour garantir la sécurité de fichiers confidentiels, tous les fichiers ASCII sont encodés, à partir de la clef de lecture (les fichiers sans clefs n'étant pas encodés). De ce fait, la clef de lecture ne peut pas être modifiée (sauf en recopiant le fichier) sous peine de ne pouvoir relire le fichier.

## LE BASIC-DOS

Le BASIC DOS est une version étendue de l'interpréteur BASIC Proteus III. Tout en continuant à utiliser la RON-BASIC, il comporte un additif en RAM chargé depuis le disque dans les RAM-SYSTEME, situées en dehors des 16 ou 32 K octets utilisateur.
Le BASIC-DOS est appelé grâce à la commande BASIC de DOS et s'emploie de façon identique au BASIC-PROTEUS III en ROM ; il possède cependant un certain nombre de fonctions supplémentaires, les plus importantes étant :

- **OPEN.** Permettant d'ouvrir et, le cas échéant, de créer un fichier.
- **CLOSE.** Permettant de fermer, et éventuellement de détruire un fichier.
- **GET** et **PUT.** Permettant d'écrire et de lire *en binaire* un fichier.
- **INPUT** et **PRINT.** Permettant d'écrire et de lire *en ASC II* un fichier ou un périphérique.
- **REWIND.** Permettant de se positionner en tête d'un fichier.
- **SEARCH.** Permettant de se positionner devant un enregistrement spécifié par son numéro ou par une clef.
- **TIME.** Permettant de connaitre l'heure en nombre de secondes depuis minuit.
- **TIME.** Permettant de disposer d'une chaine de caractères donnant l'heure sous la forme HH/MN/SEC.
- **DATE.** Permettant de disposer d'une chaine de caractères donnant la date sous la forme JJ/MM/AA.

*Par ailleurs, un certain nombre de fonctions du* BASIC PROTEUS III *ont été améliorées.*

**PROTEUS PRINT**

Imprimante sur papier normal (non métallisé), travaille sur 80 colonnes - 1.200 Bauds (120 ch./s). Avec cordon.

**Prix** ........................ **10758 F**

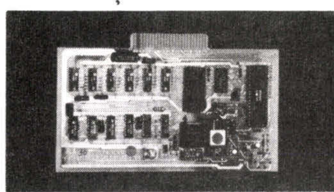
**PROTEUS PRINT MOD. 43C**

Imprimante 132 colonnes, 300 Bauds, Matrice 7 x 9 - RS 23Z. Clavier standard ASCII.

**Prix avec cordon** ............ **12466 F**

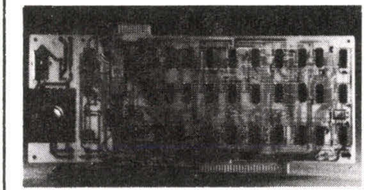
# DEUX SYSTÈMES DE VISUALISATION D'INFORMATIONS SUR TERMINAL VIDÉO

## Une carte pour voir, à la française...



Cette carte est un outil de travail professionnel. L'entrée des informations se fait en ASCII, en série ou en parallèle. La mémoire de l'écran se trouve sur la carte, ainsi que la gestion de celui-ci. La sortie est en vidéo directe, c'est-à-dire qu'elle se branche directement sur l'entrée vidéo de votre téléviseur, ou, sur l'entrée UHF par l'intermédiaire d'un modulateur. Elle gère 16 lignes de 64 caractères en vitesse réglable jusqu'à 12.000 bauds en série.
Carte interface vidéo SFF 96364, **1512 F** en kit, avec notice en français.

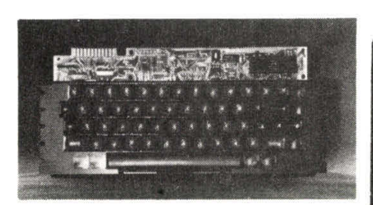
## Une carte pour voir à l'américaine...



Cette carte a été conçue par MOSTEK aux U.S. et modifiée pour fonctionner sur le standard français. Elle est commercialisée montée et testée. Elle possède un avantage évident pour les radio-amateurs : elle fonctionne en Baudot à 47,5 bauds et peut faire la conversion ASCII/Baudot. L'entrée des informations se pratique en série ou en parallèle, de plus, le curseur est adressable en absolu ou en relatif ce qui permet un pseudo-graphisme.
*Carte interface vidéo MK 3870,* **1580 F** *montée.*

## Quand la force de frappe devient caresse...

*Clavier ASCII, dernier-né de KEYTRONIC* **980 F**.



Nous vous présentons un clavier aussi moderne que votre microprocesseur. Malgré la technique employée (pas de contacts) et la sophistication de sa gestion, il est vendu à un prix très compétitif. Il devient très vite lassant, en cours de programmation de taper sur un clavier qui rebondit ou dont les touches manquent de souplesse. Sa sortie s'effectue sur 7 bits + strobe et il s'alimente uniquement en 5 volts. Il est géré par un microprocesseur GI spécialisé pour ce travail.

# PENTASONIC



☎ 524-23-16
**SUR LE PONT DE GRENELLE**
**5, rue Maurice-Bourdet - 75016 PARIS**
A 50 mètres de la Maison de la Radio
Autobus : **70-72** (arrêt MAISON DE L'ORTF). MÉTRO : **Charles-Michels**

☎ 331-56-46
**AUX GOBELINS**
**10 boulevard Arago - 75013 PARIS**
*Métro : Gobelins*

# Editorial

Si pour l'ensemble des pays européens les microprocesseurs ont fait leur grande apparition auprès du public durant l'année 1978 avec la sortie presque simultanée de revues françaises, anglaises, allemandes et italiennes, il est intéressant de regarder à quel point ce phénomène ne cesse de préoccuper les pouvoirs publics quant à ses retentissements économiques, en France et à l'étranger.

C'est ainsi que notre pays s'est doté d'un « **Plan circuits intégrés** » qui apportera sur cinq ans une aide de 600 millions de francs pour le développement de quatre pôles industriels de fabrication avec : Thomson C.S.F., division Sescosem dans le domaine des circuits bipolaires linéaires ; RTC, filiale française du groupe Philips, dans le domaine des circuits bipolaires logiques rapides ; EFCIS, filiale commune du CEA et de Thomson-C.S.F., dans le domaine des circuits MOS (« P », « C », « S.O.S. », « N ») ; et pour terminer Saint-Gobain – Pont-à-Mousson et National-Semiconductor avec la création d'une nouvelle société dans le domaine des circuits « N-MOS » et « C-MOS ».

De leur côté, nos voisins britanniques se montrent plus ambitieux ne serait-ce qu'en injectant sur trois ans l'équivalent de 1,4 milliards de francs. D'autre part, considérant qu'aux Etats-Unis le développement de la micro-électronique a créé 700 000 emplois dans la « **Silicon-Valley** » de Californie, le Premier ministre anglais, **M. Callaghan,** tout en faisant allusion à ses problèmes de chômage, conclut que son pays peut trouver là l'occasion de devenir la « Silicon Valley » de l'Europe.

Pour ce qui est de la France, les espoirs sont en grande partie fondés sur la *« Vallée grenobloise »* où jusqu'à présent la société **EFCIS** s'est essentiellement consacrée avec succès à la fabrication de circuits intégrés à la demande. Tous ces efforts visent à doter notre pays d'une indépendance relative en matière d'approvisionnement.

Il faut ajouter à ces actions d'aide à la fabrication de circuits intégrés, des actions d'information et de formation à l'utilisation de ceux-ci. Il s'agit, en effet, d'inciter et d'aider les PMI à recourir aux solutions micro-électroniques dans la conception de nouveaux produits ou dans la rénovation de produits existants.

De très nombreuses professions peuvent trouver un essor nouveau en intégrant dans leurs produits les apports de l'électronique. (Ex. : pesage, automobile, jouet, électroménager...). Pour le moment les techniciens de ces professions ne disposent pas encore d'une formation suffisante propre à leur assurer une parfaite maîtrise de composants tels que les microprocesseurs. Un vaste effort sera entrepris à cet égard.

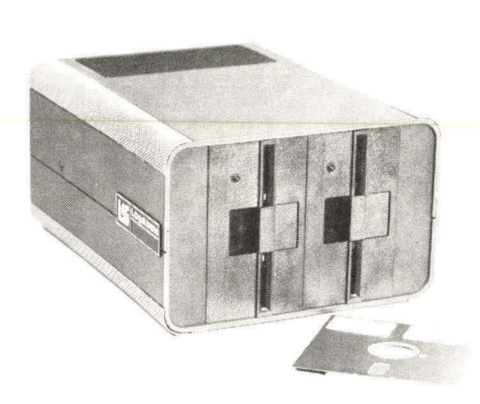
Les conclusions d'une étude faite à ce sujet ont révélé qu'il ne suffit pas seulement d'offrir des stages de formation pour qu'une société jusque là fermée à l'électronique puisse du jour au lendemain se lancer, par exemple, dans la réalisation de serrures électroniques alors que tout son savoir-faire réside dans le travail des métaux.

Il faudra pour cela implanter des structures « relais » dont la vocation sera de proposer aux PMI, non familiarisées avec la micro-électronique, les services nécessaires à l'introduction de celle-ci dans leurs produits en vérifiant la validité de l'option tant du point de vue technique qu'économique, en contrôlant le développement de la partie électronique du produit et en assurant les relations entre l'entreprise et le fabricant de circuits intégrés. De telles sociétés de service existent déjà aux Etats-Unis. En France ce besoin est nouveau et il ouvre de ce fait un terrain propice à la création d'entreprises pour lesquelles nombre de nos ingénieurs possèdent toutes les compétences nécessaires. De telles initiatives seront vivement encouragées par le ministère de l'Industrie et il est permis de supposer qu'une aide financière pourrait venir épauler ces entreprises dans leurs débuts.

D'ores et déjà, on peut dire que si 1978 a consisté pour le public en une prise de conscience du phénomène « microprocesseur », 1979 verra apparaître au grand jour l'utilisation effective des microprocesseurs et micro-ordinateurs entraînant avec eux la naissance de nouvelles entreprises parmi lesquelles ces sociétés « relais » ne constituent qu'une phase initiale. ■

**J.-J. WANEGUE.**

# L'ordinateur personnel français.



Une technologie maîtrisée, la volonté permanente d'innover et la connaissance approfondie des besoins en informatique des entreprises et des individus ont permis à LOGABAX de mettre au point le premier ordinateur personnel français : le LX 500.

Compact, d'un prix modique eu égard à ses capacités et ses performances, facilement utilisable par des non spécialistes dans leur cadre professionnel, le LX 500 se présente dès aujourd'hui comme une famille de produits;

**- LX 510** - 11.000 F H.T.* - constitue la version de base : une unité centrale à microprocesseur, 1,5 K octets de mémoire morte (ROM), 16 K octets de mémoire vive (RAM), 2 entrées/sorties aux normes V-24 du CCITT, une unité de mini-disque souple, disquette de 5 1/4 pouces, capacité 90 K octets.

**- LX 515** - 14.000 F HT* - Système comprenant une deuxième unité de disque souple : capacité de la mémoire auxiliaire portée à 180 K octets.

**- Extension de la mémoire vive de 16 K octets**, portant la capacité totale de mémoire interne à 32 K octets - 3.000 F H.T.*.

**- LX 600** - 9.600 F H.T.* - Terminal clavier imprimante, clavier ASCII, imprimante thermique à matrice 5 x 7, 80 colonnes, vitesse 30 cps.

La famille LX 500 dispose d'un logiciel complet comprenant un système d'exploitation BDOS permettant les fonctions fondamentales nécessitées par la présence d'un disque et d'un langage de programmation : le BASIC.

L'initialisation automatique dès la mise sous-tension libère l'utilisateur d'un dialogue complexe avec le système et lui permet de se consacrer exclusivement à l'application.

*Prix valables au 1/11/78.

Synergie, K.E. Création ENFI Design - Photo SIP.

**LogAbax**
*informatique*

## Premier constructeur français de mini et péri-informatique.

Bureau de Vente, 146 Av. des Champs-Élysées - 75008 Paris. Tél. 359 61 24

# toute une ligne informatique...
# chez un même constructeur

## c'est la garantie d'avoir un ensemble cohérent

**EN KIT ou EN ORDRE DE MARCHE... CHOISISSEZ !**





### H8 MICRO 8 Bits avec 8080 A

- Extensible jusqu'à 56 Ko. Horloge 2 Mhz.
- Panneau avant intelligent, terminal dynamique incorporé.
- Bus rapide - Logiciel étendu : DBUG, Editeur de texte, Assembleur, BASIC étendu, DOS.

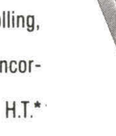
A PARTIR DE ............. **3.440 F** H.T.*





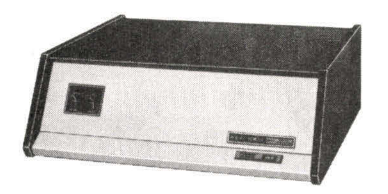
### TERMINAL VIDEO H9





- Mode conversationnel ou par lots.
- ASCII - 67 touches - page mémoire.
- 80 CAR./12 lignes, ou 20 CAR./12 lignes sur 4 colonnes.
- Plotting, semi-graphique, auto-scrolling, matrice 5 x 7.
- Interface standard série et parallèle incorporées.

PRIX .................. **4.240 F** H.T.*



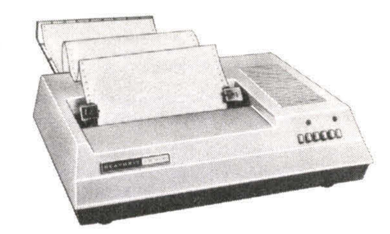
### H11 A MINI 16 Bits LSI 11/2

- Equivalent du PDP 11/03, et entièrement compatible.
- 8 x 16 bits registres, 400 instructions.
- RAM extens. à 60 Ko, Horloge 10 Mhz.
- Logiciel étendu : Assembleur, BASIC, Focal, Fortran.

A PARTIR DE .............. **7.900 F** H.T.*

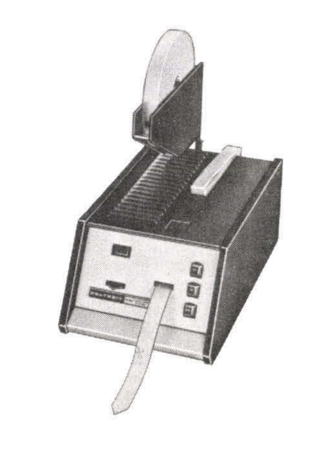


### IMPRIMANTE 165 CPS H14



- Matrice 5 x 7, 96 CAR.ASCII (majuscules et minuscules).
- Papier ordinaire, entraînement par picots.
- 80 à 132 colonnes, espacement hard ou soft.

A PARTIR DE ............ **3.220 F** H.T.*



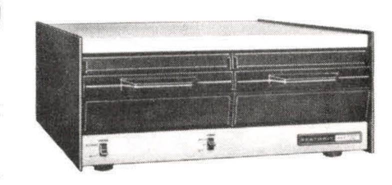
### H10 LECTEUR PERFORATEUR

- Lecteur 50 CPS - Perforateur 10 CPS.
- Interface parallèle TTL standard.
- Dispositif de copie interne.

PRIX .................. **2.516 F** H.T.*

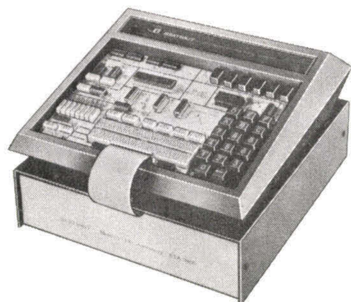


### MINI-DISQUES pour H 11 H27



- Compatible av. DEC RX11, géré par Z 80.
- 2 disques Memorex - 512 Ko sectoré soft ou hard.
- Possibilité format IBM 3740.
- DOS étendu : Edit, BASIC, Fortran, Assembleur.

A PARTIR DE ........... **11.900 F** H.T.*



### EE 3401 MICRO 8 Bits avec 6800

- Trainer pour expérimentation.
- Extension RAM, Interfaces, BASIC.
- Cours complet sur microprocesseurs.

### EC 1100 COURS DE BASIC

- Auto-éducation permanente.



### MINI-DISQUETTES pour H 8 H17



- 1 ou 2 lecteurs WANGCO.
- Simple face, simple densité.
- Hard sectoré - 102 Ko/disque.
- Pas entre piste 30 ms
- DOS étendu : Edit, Assembleur, DBUG, BASIC, Adressage direct.

A PARTIR DE ............. **3.986 F** H.T.*

* Prix en Kit (H.T.) au 1/02/79

**BON A DECOUPER, à adresser à**

**HEATHKIT**
**Schlumberger**

FRANCE : HEATHKIT, 47 rue de la Colonie, 75013 PARIS, tél. 588.25.81
BELGIQUE : HEATHKIT, 16 av. du Globe, 1190 BRUXELLES, tél. 344.27.32

**Je désire recevoir votre catalogue** couleur en Anglais - Je joins 2 timbres à 1,20 F pour frais d'envoi.

Nom, prénom ..........................................................................

Adresse ..............................................................................

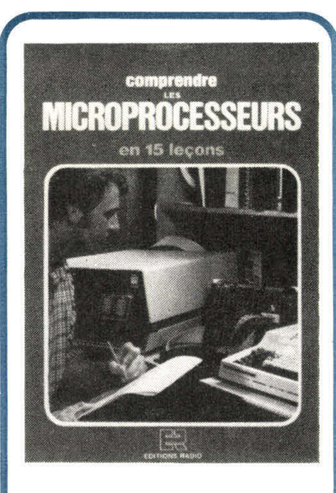
M.S. 03-79

CENTRES
DE DEMONSTRATION

**PARIS** (6e) 84 bd. Saint-Michel
Téléphone : 326.18.91

**LYON** (3e) 204 rue Vendôme
Téléphone : (78) 62.03.13

# Microprocesseurs : revue de la littérature

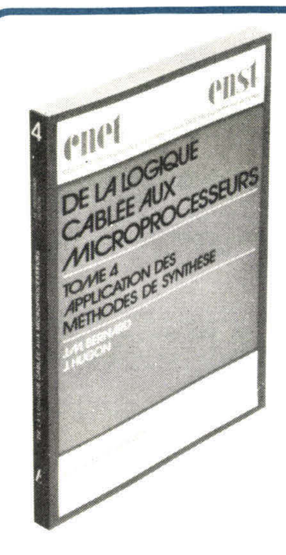



### COMPRENDRE LES MICROPROCESSEURS
*(en 15 leçons)*

**Auteur : Daniel Queyssac.**
**Editeur : Les Editions Radio.**
**Format : 14 x 20 cm.**
**Nombre de pages : 160.**
**Prix : 29 F.**

1 – Qu'est-ce qu'un microprocesseur.
2 – Le système binaire.
3 – Conception d'un système microprocesseur de base.
4 – Le matériel.
5 – Techniques d'entrées/sorties.
6 – Comment résoudre un problème d'entrée/sortie.
7 – Exemple d'application entrée/sortie.
8 – Introduction au concept de programmation des microprocesseurs.
9 – Modes d'adressages de la mémoire du MPU.
10 – L'utilisation des outils de support logiciel.
11 – Langage de programmation des microprocesseurs.
12 – Techniques de programmation.
13 – Section et choix du MPU.
14 – Processus de l'étude d'un système.
15 – Le marché des microprocesseurs.

*Ce livre, plus particulièrement destiné aux débutants, étudiants et techniciens, contient tout ce qu'il faut pour accéder à ces techniques.*




### DE LA LOGIQUE CABLEE AUX MICROPROCESSEURS
*(4 tomes)*

**Auteurs : J.-M. Bernard ; J. Hugon et R. Le Corvec (Ingénieurs au CNET).**
**Editeur : Eyrolles. Collection Technique et Scientifique des Télécommunications.**
**Format : 16 x 24 cm.**
**Nombre de pages et prix : Tome 1 : 212 p., 234 fig., 93 F. – Tome 2 : 136 p., 141 fig., 64 F. – Tome 3 : 164 p., 94 fig., 76 F. – Tome 4 : 268 p., 178 fig., 120 F.**

**Tome 1 :** Circuits combinatoires et séquentiels fondamentaux : Notation binaire, fonctions logiques. Les opérateurs logiques élémentaires. Les familles logiques. Fonctions combinatoires usuelles. Représentation des nombres codés. Opérations arithmétiques et circuits associés. La fonction de mémorisation. Décalage et comptage. Les bascules RS Maître-esclave, JK et D. Réalisation des compteurs. Les mémoires. Représentation symbolique des fonctions logiques.

**Tome 2 :** Applications directes des circuits fondamentaux : Simplification des fonctions logiques. Etude et assemblage des circuits combinatoires. Synthèse des fonctions logiques. Changement de code. Les additionneurs. Les registres compteurs. Le décalage. Gestion de mémoire en pile ou en file.

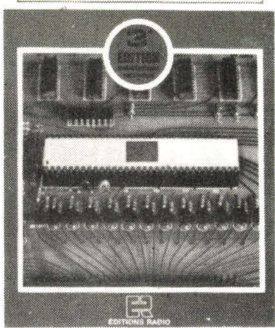
**Tome 3 :** Méthodes de conception de systèmes : Découpage temporel. Systèmes séquentiels. Séquenceur câblé et méthode de synthèse. Séquenceur microprogrammé. Anomalies fréquentes dans les schémas. Caractéristiques des microprocesseurs. Synthèse avec microprocesseur. Réalisation des équipements. Comparaison entre méthodes.

**Tome 4 :** Application des méthodes de synthèse : Cycles d'horloge. Organigrammes fonctionnels. Séquenceurs câblés. Gestion d'un terminal. Etude d'un schéma non structuré. Gestion d'un système de télémesure. Etude du microprocesseur 6800. Réalisation du système de télémesure avec microprocesseur. Utilisation d'un micro-ordinateur.

*Le tome 1 et le tome 2 présentent les notions de base sur les circuits combinatoires et séquentiels. Ces deux tomes s'adressent aux débutants ne possédant aucune expérience en logique.*

*Le tome 3 décompose un système numérique en 2 parties : le séquenceur qui fournit des commandes et les applications.*

*Le tome 4 illustre par des exemples l'emploi des différentes méthodes de réalisation d'un séquenceur.*



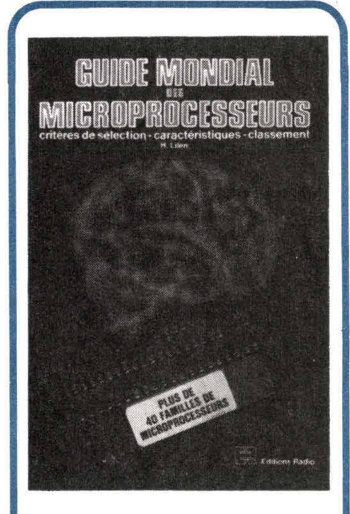

### DU MICROPROCESSEUR AU MICRO-ORDINATEUR

**Auteur : H. Lilen.**
**Editeur : Les Editions Radio.**
**Format : 16 x 24 cm.**
**Nombre de pages : 448 pages, 320 illustrations.**
**Prix : 95 F.**

Introduction à la micro-informatique. Les circuits intégrés. Les opérations arithmétiques. Les opérations logiques. Les mémoires. Fonctionnement de l'unité centrale. Les entrées-sorties. Vers les sytèmes réels. Les périphériques. Le logiciel. Classement des microprocesseurs. Analyse de microprocesseurs PMOS. Analyse de microprocesseurs NMOS. Microprocesseurs CMOS. Microprocesseurs bipolaires. Exemples de micro-ordinateurs. Les sytèmes d'évaluation et de développement. Systèmes multi-microprocesseurs et réseaux. Applications industrielles et informatiques. Applications au grand public, aux télécommunications et à l'instrumentation.

*Certainement un des ouvrages les plus complets édités sur l'ensemble des questions relatives à la technologie, la réalisation et la programmation de systèmes à microprocesseur.*

# Microprocesseurs : revue de la littérature

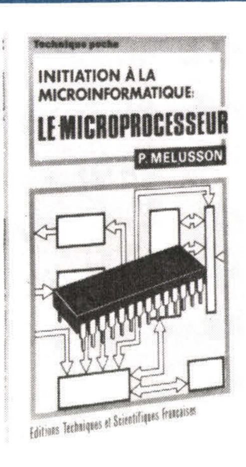



**GUIDE MONDIAL DES MICROPROCESSEURS**

**Auteur : H. Lilen.**
**Editeur : Les Editions Radio.**
**Format : 21 x 29,7 cm.**
**Nombre de pages : 192.**
**Prix : 95 F.**

— Etude des critères de sélection et conseils sur le choix d'un microprocesseur.
— Classements par terminologies et fabricants ainsi que par longueurs de mots.
— Liste des secondes sources possibles.
— Condensé de la documentation des constructeurs présents sur le marché et correspondance des termes utilisés.
— Adresses des fabricants et distributeurs.

*Destiné aux utilisateurs de microprocesseurs, cet ouvrage les guidera dans leur choix des circuits.*

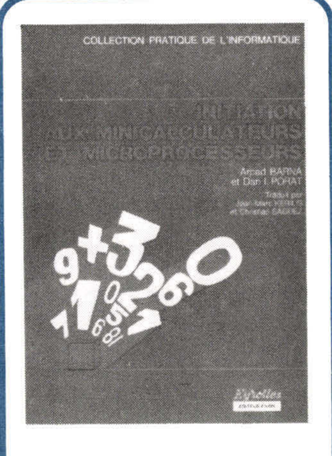



**INITIATION À LA MICRO-INFORMATIQUE : LE MICRORPOCESSEUR**

**Auteur : P. Mélusson.**
**Editeur : Editions Techniques et Scientifiques Françaises (E.T.S.F.).**
**Format : 12 x 16,5 cm (Technique poche).**
**Nombre de pages : 138.**
**Prix : 27 F.**

1 - Introduction.
2 - Le cerveau humain et l'ordinateur : cerveau robot : Le Hardware et le Software. Différences entre l'homme et la machine.
3 - Ordinateur - Calculateur.
4 - Les langages d'ordinateurs - Les appareils de décodage - Des langages en « binaire pur » : Tableau des principaux langages de programmation.
5 - Le calcul binaire - Les codages : Octal, Hexadécimal et BCD : Tableau des correspondances. Conversion décimal/binaire et binaire/décimal. Les 4 opérations en système binaire. Conversion programme-source/programme-objet.
6 - Les fonctions logiques : Les fonctions logiques combinatoires et les séquentielles.
7 - La technologie des microprocesseurs.
8 - Organisation des microprocesseurs.
10 - Les circuits et systèmes d'interface entre microprocesseur mémoire unité centrale et périphériques.
11 - La programmation.

*Ouvrage d'initiation à la microinformatique et au microprocesseur. Utile pour tous ceux désirant acquérir les bases de cette technique.*

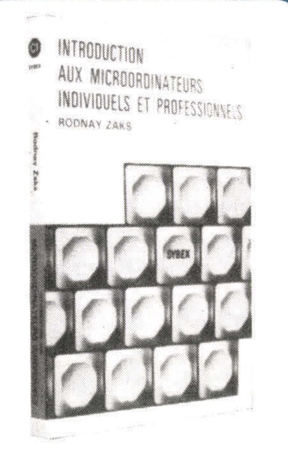



**INITIATION AUX MINI-CALCULATEURS ET MICROPROCESSEURS**

**Auteurs : A. Barna et D.I. Porat.**
**Editeur : Eyrolles collection « Pratique de l'Informatique ».**
**Format : 15,4 x 22 cm.**
**Nombre de pages : 118 ; 38 figures.**
**Prix : 49 F.**

Structure de base des minicalculateurs et microprocesseurs : Bloc entrée/sortie - L'unité centrale - La mémoire - Les microprocesseurs.
Instructions en langage machine, en langage d'assemblage - Langages évolués de programmation - Sous-programmes - Organigrammes. Entrées et sorties : Instructions d'entrée-sortie - Interruptions - Accès direct en mémoire. Opérations arithmétiques.
- Représentation en virgule flottante et règles de calcul. Circuits arithmétiques et logiques : accumulateur et unité arithmétique et logique. La mémoire centrale : Structure - Registre à décalage - Registres auxiliaires - Circuits de restauration pour les RAM MOS dynamiques - Modes d'adressage - Adressage indirect. L'unité de commande : Microprogrammation - Schéma fonctionnel d'un mini-calculateur. Compléments : Assembleurs - Chargeurs - Structures des données - Liens de sous-programme - Simulation - Partage hardware.

*Ce livre s'adresse aux personnes qui n'ont pas de connaissances particulières dans ces domaines mais qui désirent atteindre un niveau suffisant leur permettant une utilisation intelligente des mini-calculateurs et microprocesseurs.*



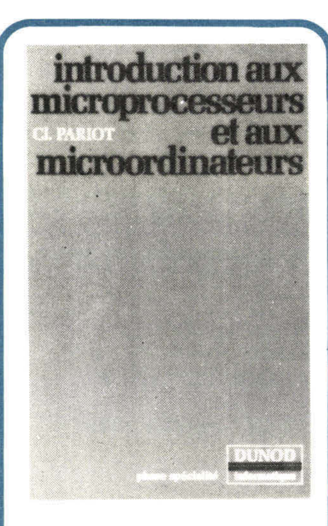

**INTRODUCTION AUX MICRO-ORDINATEURS INDIVIDUELS ET PROFESSIONNELS**
*Réf. : C1*

**Auteur : Rodnay Zaks.**
**Editeur : Sybex.**
**Format : 14 x 21 cm.**
**Nombre de pages : 280 ; 125 illustrations.**
**Prix : 52,43 F.**

1 - L'ère du micro-ordinateur.
2 - Comment l'utiliser.
3 - Définitions de base : applications, fabrication, UC, bus, mémoire, entrées-sorties, système.
4 - Fonctionnement : assemblage d'un système.
5 - Programmation : langages de programmation, organigrammes, représentations binaires.
6 - BASIC et APL : boucles, instructions, fonctions, sous-programmes.
7 - Informatique commerciale : traitement de mot. Automation de comptabilité, de gestion. Fichiers adresses. Programmes commerciaux.
8 - Choix d'un système.
9 - Les périphériques : Clavier, CRT, imprimantes, disques, cassettes, autres.
10 - Choix d'un micro-ordinateur.
11 - Coût d'un système.
12 - Comment échouer : Défaillances du matériel - du logiciel - Amélioration de la fiabilité - Le « choc-ordinateur ».
13 - Secours : Où le trouver : publications, clubs, points de vente, formation.
14 - Demain.

*Cet ouvrage s'adresse à ceux qui souhaitent comprendre les micro-ordinateurs et peut-être en acquérir un. Il ne requiert aucune formation technique préalable.*

# Microprocesseurs : revue de la littérature

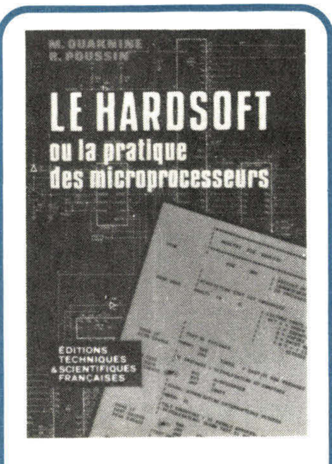



### INTRODUCTION AUX MICROPROCESSEURS ET AUX MICRO-ORDINATEURS

**Auteur : Claude Pariot (ingénieur à IBM-France).**
**Editeur : Collection « Dunod Informatique ».**
**Format : 15,5 x 24 cm.**
**Nombre de pages : 144.**
**Prix : 69 F.**

1 – Rappels généraux sur le fonctionnement des ordinateurs.
2 – Présentation des microprocesseurs et des micro-ordinateurs.
3 – Technologie des microprocesseurs et des micro-ordinateurs.
4 – Description des microprocesseurs et des micro-ordinateurs.
5 – Fonctionnement des microprocesseurs et des micro-ordinateurs.
6 Utilisation des microprocesseurs et des micro-ordinateurs.
7 – Applications des microprocesseurs et des micro-ordinateurs.
8 – Inventaire des microprocesseurs. Annexes. Bibliographie. Index.

*Par son orientation pédagogique et grâce à un chapitre introductif sur les ordinateurs, cet ouvrage s'adresse aux étudiants, enseignants et techniciens qui ont à se familiariser avec les microprocesseurs.*

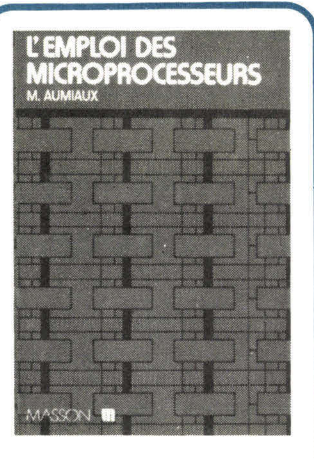



### LE HARDSOFT OU LA PRATIQUE DES MICROPROCESSEURS

**Auteurs : M. Ouaknine et R. Poussin.**
**Editeur : Editions Techniques et Scientifiques Françaises (E.T.S.F.).**
**Format : 15 x 21 cm.**
**Nombre de pages : 200.**
**Prix : 65 F.**

Structure d'un ordinateur.
L'unité centrale : Unité arithmétique et logique - Unité de commande – Logique des interruptions – Le mécanisme d'entrées-sorties.
Les familles de microprocesseurs : Principes généraux – L'évolution – Choix d'un microprocesseur.
L'unité centrale du 8080 .
La mémoire centrale.
Coupleurs : Coupleur 8 bits – Interface parallèle programmable – Interface série programmable.
Les différents stades.
Les outils.
Les techniques : Sous-programme – Table de branchement – Macro-instructions – Gestion des interruptions.
Programmes arithmétiques : Multiplication – Division – Addition multibyte – Soustraction multibyte.
Système de base.
Compte-tours digital intelligent.
Système industriel : Installation de trempe.
Un microsystème : le 8048.
La 3e génération : le Z80.
Le 8086 : la 4e génération ?.

*Ce livre, destiné à l'étudiant et au technicien, aborde principalement l'unité centrale du 8080, les nouveaux circuits micro-ordinateur en un boîtier et les 16 bits.*

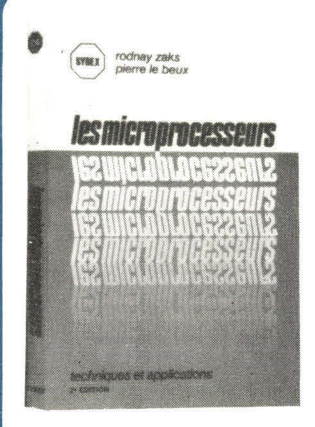



### L'EMPLOI DES MICROPROCESSEURS

**Auteur : Michel Aumiaux (Ingénieur E.S.E.O.).**
**Editeur : Masson.**
**Format : 16 x 24 cm.**
**Nombre de pages : 128 ; 80 figures.**
**Prix : 60 F.**

1 – Notions de base préliminaires : Codes – Fonctions logiques et opérateurs – Bascules, registres, compteurs, décodeurs – Logique trois états et bus – Mémoire.
2 – Architecture et Exploitation d'un système à microprocesseur : Logique câblée et logique programmée – Matériel (hardware) et interfaces – Le logiciel (software) – Le système de développement : architecture – Logiciel – Possibilités.
3 – Principes généraux pour l'exploitation d'un système à microprocesseur .
4 – Etude du matériel d'un système à microprocesseur : Liaisons processeur-mémoire et microprocesseur-inferface-périphérique.
5 – Le logiciel : Programmes logés dans la RAM. Principes de programmation conseillés. Exercices d'application. Utilisation d'un moniteur.

*L'objectif de cet ouvrage est de situer et définir les systèmes logiques à microprocesseurs, d'étudier les constituants de ces systèmes et la façon de les réaliser concrètement, puis d'appliquer ces connaissances théoriques à l'étude concrète du point de vue de l'utilisateur.*




### LES MICROPROCESSEURS (DU COMPOSANT AU SYSTÈME)

*Réf. C4*

**Auteurs : Rodnay Zaks et Pierre Lebeux.**
**Editeur : Sybex.**
**Format : 16 x 24 cm.**
**Nombre de pages : 320 ; 200 illustrations.**
**Prix : 95,23 F TTC.**

1 – Concepts de base. Principes de fonctionnement – Les bus – La mémoire – Fabrication d'un microprocesseur – Les technologies LSI.
2 – Fonctionnement interne d'un microprocesseur : Architecture – Etude de cas : le 8080.
3 – Les composants d'un système : Les mémoires (ROM, RAM). techniques et boîtiers d'entrée-sortie (UART, PIO) – Interruptions, DMA.
4 – Une évaluation comparée des microprocesseurs : Classification. Les 4 bits, 8-bits, 16-bits. – Microordinateurs en un boîtier – Tranches de bit.
5 – Interconnexion d'un système : Architecture – Assemblage d'un CPU – Connexion.
6 – Applications des microprocesseurs .
7 – Techniques d'interface : Clavier – LED – Télétype – Disque souple – Ecran CRT – Multimicroprocesseurs – Bus standard.
8 – Programmation : Types d'instruction – Langage assembleur – Multiplication – Logique programmée.
9 – Développement d'un système.
10 – Le futur.

*Ouvrage de base sur les microprocesseurs pour toute personne ayant une formation technique ou scientifique. Il s'agit d'un livre conçu pour la formation qui se lit facilement malgré sa technicité.*

# Microprocesseurs : revue de la littérature

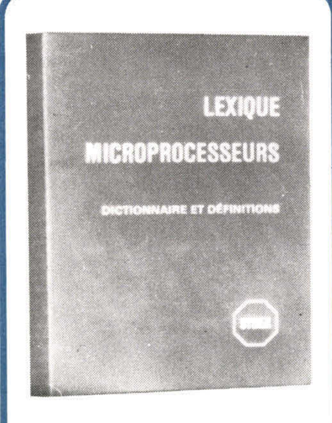



**LES SYSTÈMES MICROPROGRAMMÉS**
*(automates, mini et microprocesseurs)*

**Auteurs : Th. Maurin et M. Robin.**
**Editeur : Dunod Technique.**
**Format : 15,5 x 24 cm.**
**Nombre de pages : 140.**
**Prix : 49 F TTC.**

1 – Démystification des microprocesseurs.
2 – Les fondements de la logique programmée.
3 – Support matériel de la logique programmée : bascules et mémoires.
4 – Exercices et synthèse de systèmes combinatoires et séquentiels.
5 – Automates programmables.
6 – L'unité logique et arithmétique : organe de traitement, le circuit SN74131.
7 – L'unité de contrôle et de traitement : architecture.
8 – Macroprogrammation : sous-programme.
9 – Structure du microprocesseur 8080 d'Intel : structure d'une carte et programmation.
10 – Exercices : code opération et mnémonique.

*Approche très pédagogique qui permet un exposé progressif et suit l'évolution historique conduisant les logiciens des portes intégrés aux microprocesseurs.*




**LEXIQUE MICROPROCESSEURS**
*(Réf : C2)*

**Editeur : Sybex.**
**Format : 16 x 12 cm.**
**Nombre de pages : 120.**
**Prix : 19,80 F TTC.**

Dictionnaire anglais-français. Lexique (définitions). Numéros de composants. Les signaux du bus S100, de RS232C, de IEEE 488. Les fabricants de microprocesseurs. Définitions militaires. Conversion décimale, binaire, hexadécimale, octale.

*Livre de poche contenant la traduction des termes usuels en anglais et leur définition en français, ainsi que les abréviations du jargon microprocesseur.*

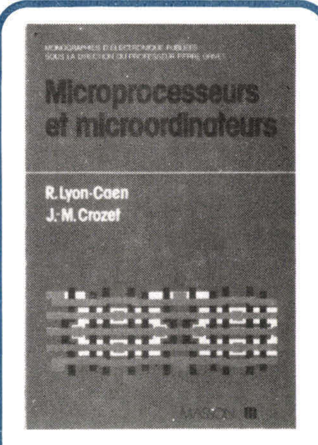



**L'ORDINATEUR ET L'INFORMATIQUE**
*(en 15 leçons)*

**Auteur : P. Morvan.**
**Editeur : Les Editions Radio-Hachette.**
**Format : 14 x 20 cm.**
**Nombre de pages : 226.**
**Prix : 30 F.**

1 – A la rencontre de l'information.
2 – Où l'on cherche un supporter (différents support de données).
3 – Où les données accèdent à l'ordinateur.
4 – La sortie des résultats.
5 – Quand l'ordinateur dessine.
6 – Un peu d'électronique.
7 – A la rencontre de la machine.
8 – Programmons l'ordinateur.
9 – Quand l'homme exploite la machine.
10 – Utilisons les périphériques.
11 – Partageons notre ordinateur.
12 – La multiprogrammation.
13 – Où l'on s'en paye une tranche.
14 – Quand l'ordinateur est loin.
15 – A chacun son informatique.

*Certainement le plus simple de tous les ouvrages décrits ici. Destiné à tous ceux qui désirent accéder à l'informatique.*




**MICROPROCESSEURS ET MICRO-ORDINATEURS**

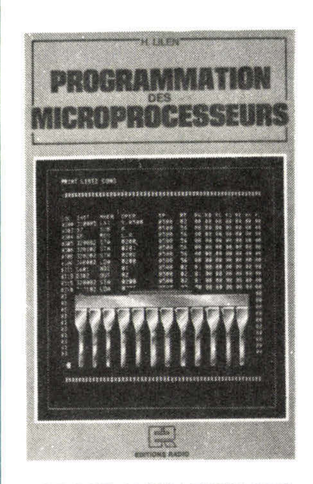
**Auteurs : Robert Lyon-Caen (professeur à E.S.E.) ; Jean-Maurice Crozet (ingénieur E.S.E.).**
**Editeur : Masson.**
**Format : 16 x 24 cm.**
**Nombre de pages : 188.**
**Prix : 60 F.**

1 – Introduction aux microprocesseurs et aux micro-ordinateurs : L'ordinateur – Le mini-ordinateur – Le micro-ordinateur, le microprocesseur.
2 – Le microprocesseur : principe, description, fonctionnement : Définition d'un microprocesseur.

3 – Etude de la logique de contrôle d'un microprocesseur : Introduction : microprocesseurs « monoblocs » et microprocesseurs « multiblocs » – Etude de la logique de contrôle sur un cas simplifié – Microprocesseur bipolaire microprogrammable : le 2900 d'AMD.
4 – Réalisation d'un micro-ordinateur autour d'un microprocesseur.
5 – Exemple d'utilisation simple d'un microprocesseur.
6 – Programmation sur micro-ordinateur logiciel d'aide à la mise au point : Rappel sur le langage machine – Langage assembleur – Programme « assembleur ». Langage « évolué ». Programme compilateur. Comparaison entre langage assembleur et langage « évolué ».
7 – Essai de classification des principaux microprocesseurs actuels.

*Livre d'un bon niveau à l'usage des électroniciens qui doivent utiliser les microprocesseurs.*

# Microprocesseurs : revue de la littérature

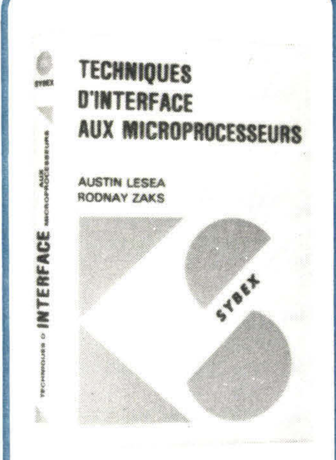



**PROGRAMMATION DES MICROPROCESSEURS**

**Auteur : H. Lilen.**
**Editeur : Les Editions Radio.**
**Format : 16 x 24 cm.**
**Nombre de pages : 240 ; 105 figures.**
**Prix : 85 F.**

Le système à microprocesseur : Définitions, fonctionnement, fonctions logiques.
Introduction à la programmation : organigramme, traduction du programme assembleur.
Modes d'adressage : Implicite, inhérent, immédiat, direct et absolu, restreint et étendu, relatif, indexé.
Langages machine et symboliques : assembleur et assemblage, octal, hexadécimal, assembleur résident et cross assembleur, syntaxe.
Instructions et directives : Format, indicateurs, jeu du 8085.
Syntace et assembleur : Les champs, les erreurs, exemples.
Ruptures et séquences : Macro-instruction, sous-programme, applications.
Les échanges d'entrée-sortie : adressage, communication, traitement.
Langages haut niveau : langages évolués et interprétés.
Les aides au développement.

*Orienté vers les applications, ce livre montre comment analyser un problème pour ensuite rédiger et mettre au point le programme approprié.*




**TECHNIQUES D'INTERFACE AUX MICROPROCESSEURS**
*Réf. : C6*

**Auteurs : Austin Lesea et Rodnay Zaks.**
**Editeur : Sybex.**
**Format : 16 x 24 cm.**
**Nombre de pages : 410.**
**Prix : 95,23 F TTC.**

1 – Introduction.
2 – Assemblage de l'unité centrale : Architecture – Adressage – Système 8080 – 6800 – Z-80 – 8085.
3 – Entrées-sorties de base : Projection sur mémoire ou sur E/S. Entrées-sorties parallèles, sérielles. Les trois techniques de gestion d'E/S : polling, interruptions, DMA.
4 – Interface aux périphériques : Claviers – Rebond, encodage, rollover. Affichages à LED. Télétype, Ruban papier – Imprimante – Lecteur de carte de crédit à piste magnétique – Magnétophone à cassettes – Ecran CRT – Composant CRTC – Disque souple FDC – CRC.
5 – Conversion analogique digitale : Circuits. Théorème d'échantillonnage, approximation successive, intégration, comparaison simultanée. Interface.
8 – Les bus standards : Parallèles : S100, S50, IEE-488, CAMAC. Sériels : EIA - RS232C, RS422, RS423. Formats synchrones.
7 – Etude de cas : un multiplexeur à 32 canaux.
8 – Mise au point et diagnostic.
9 – Conclusion - évolution.

*Cet ouvrage présente de manière progressive les concepts et techniques de base. Le niveau requis est celui du C4 (Les microprocesseurs).*

**THEORIE ET PRATIQUE DES MICROPROCESSEURS**

**Auteurs : R. Arouete et H. Lilen.**
**Editeur : Les Editions Radio.**
**Format : 16 x 24 cm.**
**Nombre de pages : 192 ; 38 illustrations.**
**Prix : 70 F TTC.**

**1re partie :** Introduction aux logiques programmées – L'information et son traitement : transmission, stockage et codage – Structure et fonctionnement des logiques programmées : le rôle du processeur, les mémoires, les entrées/sorties, les interruptions.
**2e, partie :** Introduction aux microprocesseurs : le matériel (hardware), description du microprocesseur. Le logiciel : introduction, système de développement, structuration. Conduite d'un projet.
**3e partie :** Le développement d'une étude : étude et conception d'un système à logique programmée. Réalisation pratique de l'étude : analyse et réalisation, mise au point. Exemples d'applications.

*Ouvrage général sur l'utilisation des microprocesseurs. Constitue un outil pratique qui guidera les électroniciens et les informaticiens.*

**OUVRAGES A PARAITRE**

**– PROGRAMMATION DU MICROPROCESSEUR : 6502.**

**Auteur : Rodnay Zaks.**
**Editeur : Sybex.**
**Nombre de pages : 280 ; réf. C3.**
**Prix : 95,25 F.**

Ce texte couvre pas à pas tous les aspects de la programmation. Concepts de base – Algorithme et programme – Représentations binaires – Techniques d'adressage – Programmes arithmétiques – Techniques d'entrées/sorties – Transferts programmés : sériel et parallèle – Développe- – Etudes de cas – Développement d'un programme : outils et méthodes – Systèmes à 6502 : KIM, VIM, APPLE, PET, ROCKWELL 65.

---

**– MISE EN ŒUVRE DU MICROPROCESSEUR 6800.**

**Auteurs : A. Semeteys et P. Pelloso.**
**Editeur : Infoprax (Tél. 032.06.61).**

**1re partie :**
De la logique câblée à la logique microprogrammée. Compatibilité MOS-TTL.
**2e partie :**
Généralités sur les ordinateurs orientés vers les microprocesseurs. Etude pas à pas du matériel et du logiciel.
**3e partie :**
La famille 6800. Programmation du 6800. Etude des composants de la famille. Etude détaillée d'un micro-ordinateur : le Kit MEK D2.
**4e partie :**
Logiciel d'aide à la mise en œuvre. Langage source. Assembleur. Langage machine. Systèmes de développement.

---

**– AU CŒUR DES MICROPROCESSEURS.**

**Auteurs : Dominique Girod et Roland Dubois.**
**Editeur : Eyrolles.**

# Etude détaillée d'un P.I.A. : le coupleur d'entrée/sortie M.C. 6820

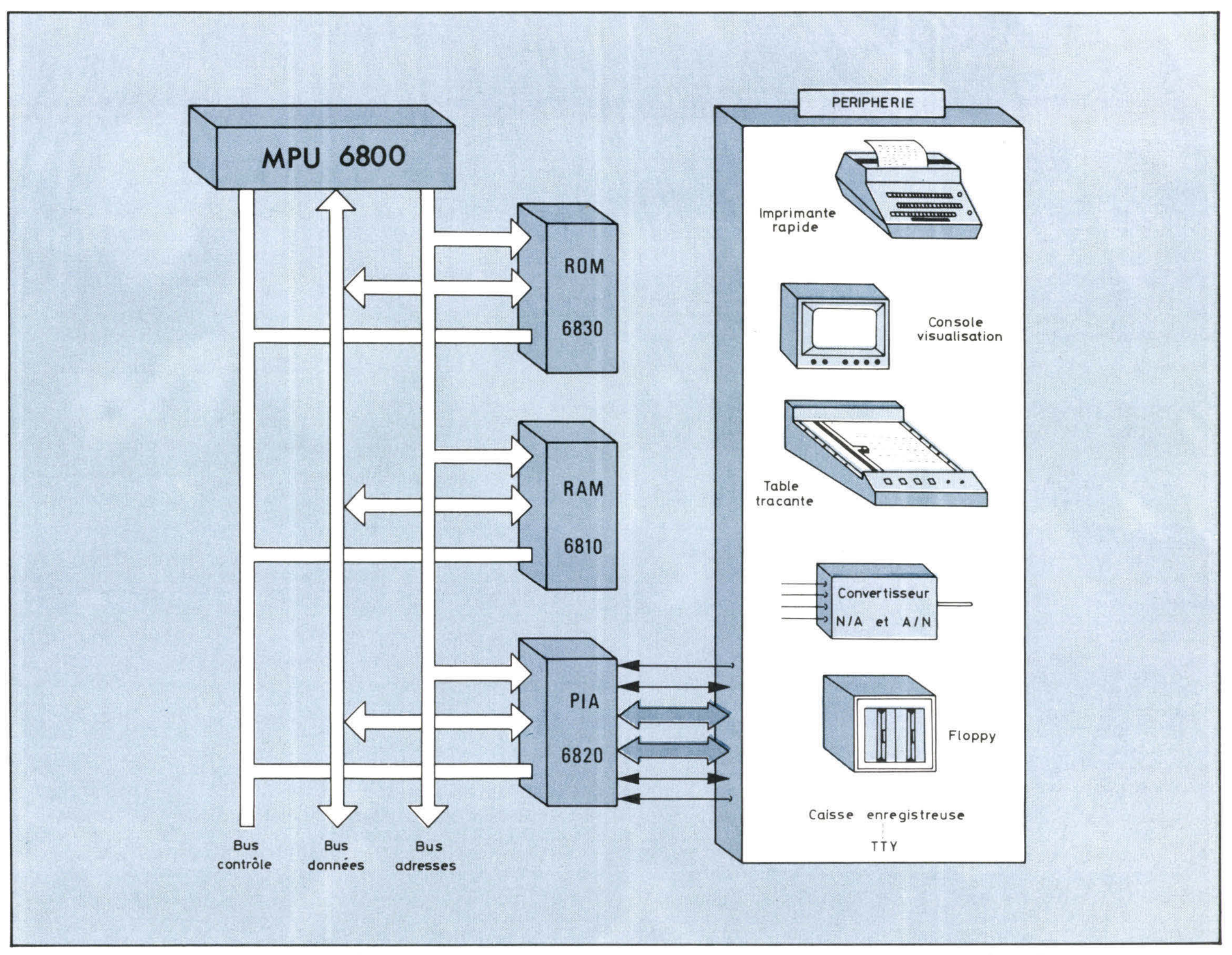



*Fig. 1. – Le coupleur d'entrée/sortie ou P.I.A. effectue la liaison entre le monde extérieur et la machine. Cette illustration montre un système construit autour des éléments de la famille 6800.*

**Tous les systèmes, quels qu'ils soient : kit d'évaluation, système de mise au point de programmes ou micro-ordinateur travaillent avec un coupleur d'entrée/sortie.**

**L'importance particulière de ce composant nous a poussé à étudier de façon détaillée l'un des modèles les plus connus et utilisés : le M.C. 6820 de Motorola ou SF.F 96821 de Sescosem.**

**Compatibles TTL, ces circuits peuvent souvent migrer d'un système microprocesseur à un autre. Ainsi, vous pouvez bénéficier des qualités inhérentes d'un coupleur déterminé en l'incorporant dans votre propre application.**

**Du point de vue hardware ces boîtiers sont très sophistiqués. Leur fonctionnement est microprogrammé et il est nécessaire de maîtriser à la perfection tous les bits de commande dont dépend leur mode de travail.**

**Conscient de l'aspect un peu ardu de leur utilisation nous avons volontairement illustré notre étude d'un grand nombre de schémas qui, nous l'espérons, vous permettront de tirer le maximum de profit de cet article.**

**Notons enfin que ces coupleurs ont plusieurs appellations différentes suivant les constructeurs PIA (Peripheral Interface Adapter) pour Motorola et Sescosem, PIO (Programmable Input/output) pour Zilog, PPI (Programmable Peripheral Interface) pour Intel et Signetics...**

# Signaux échangés

## Avec le système : PIA ↔ Système

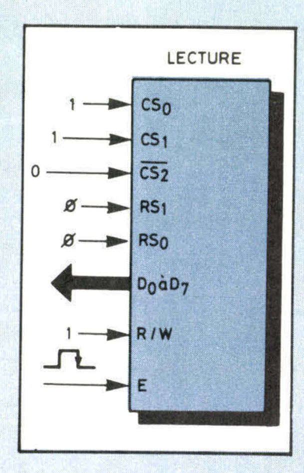
**$CS_0$, $CS_1$, $\overline{CS_2}$** — Lorsque $CS_0$, $CS_1$, $\overline{CS_2}$ = 110, le PIA est sélectionné.

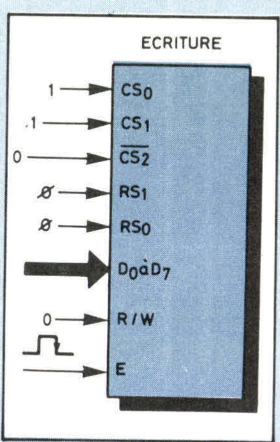
**$RS_1$, $RS_0$** — Le PIA étant sélectionné, les quatre combinaisons de ces deux bits permettent d'adresser les registres internes.

En conséquence, **le PIA occupe 4 adresses mémoire.**

**E** — Signal d'activation des échanges, généralement cette entrée est reliée à : bus $\varnothing_2$ (signal du bus contrôle)...

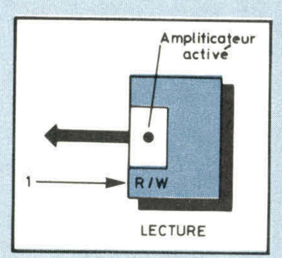
**R/W** — Signal de lecture-écriture : 1 = lecture - 0 = écriture.

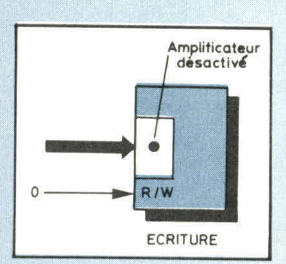



**$D_0$ $D_7$** — Bus bidirectionnel de données. Il aboutit, dans le PIA, à un amplificateur qui peut être activé ou mis dans l'état haute impédance par le signal R/W **si le PIA est sélectionné.**

Par ces 8 fils arrivent les données à transmettre en sortie (vers les périphériques) ou à lire.




**$\overline{RESET}$** — Mis à 0, ce signal remet tous les registres internes du PIA à 0.

**$\overline{IRQA}$ $\overline{IRQB}$** — 2 lignes de demande d'interruption destinées à interrompre l'exécution d'un programme par le MPU. Ces lignes sont généralement reliées aux entrées $\overline{IRQ}$ ou $\overline{NMI}$ du microprocesseur ou sont placées sur les entrées du contrôleur prioritaire d'interruptions PIC 6828.

## Avec la périphérie

**$PA_0$ à $PA_7$ $PB_0$ à $PB_7$** — 16 lignes de données programmables individuellement en entrées ou en sorties.

Ces 2 ports d'entrée/sortie reflètent, en sortie, le contenu de 2 registres internes de 8 bits dont l'état binaire apparaît sous forme de tensions de sortie (+ 5 V : « 1 » logique et O V : « 0 » logique) maintenues tant qu'il n'y a pas modification dans les registres.

**$CA_1$ $CB_1$** — 2 lignes d'entrée d'interruption.

**$CA_2$ $CB_2$** — 2 lignes programmables en entrée d'interruption ou en sortie de commande.

Dans ce dernier cas, ils reflètent directement l'état d'un bit d'un registre interne (de contrôle).

**$V_{SS}$ $V_{CC}$** — Deux bornes d'alimentation : $V_{CC}$ = + 5 V et $V_{SS}$ = 0 V.

La consommation est d'environ 110 mA et la puissance dissipée de 550 mW.

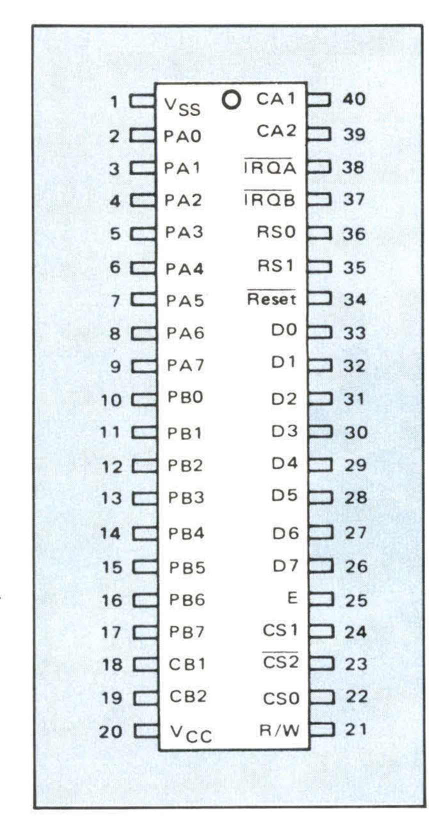



*Fig. 2. - Brochage du coupleur d'E/S 6820.*

MC 6820 est le nom du coupleur d'entrée/sortie de la famille 6800 de Motorola. Chez Sescosem, il porte la référence SF.F 96821.

Dans un système à microprocesseur, l'emplacement du coupleur d'entrée/sortie est indiqué **figure 1.** Le circuit se présente sous la forme d'un boîtier DIL à 40 broches **(fig. 2).**

Le PIA communique avec le système à l'aide des signaux représentés **figure 3.** La signification précise de chacune des broches du boîtier est explicitée de façon simplifiée dans notre encadré ci-contre.

## Organisation interne

Le schéma de la **figure 4** représente le synoptique du coupleur d'entrée/sortie.

Nous remarquons que le microprocesseur peut adresser 6 registres en écriture et en lecture.

Ces registres sont répartis en deux groupes de trois registres relatifs à chacun des borniers :

**C R A**
**C R B** Contient les paramètres de fonctionnement.

**D D R A**
**D D R B** Contient le mot fixant le sens de transfert (entrée ou sortie) pour chacune des lignes de données. Un état « 1 » définit une broche en sortie et un état « 0 » une broche en entrée.

**O R A**
**O R B** Mémorise les données en sortie lors d'une écriture. A la même adresse on peut lire les données présentées en entrée, mais elles devront être mémorisées à l'extérieur.

Deux circuits de commande d'interruption A et B permettent de traiter $CA_1$, $CA_2$, $CB_1$, $CB_2$ et de générer $\overline{IRQA}$ et $\overline{IRQB}$.

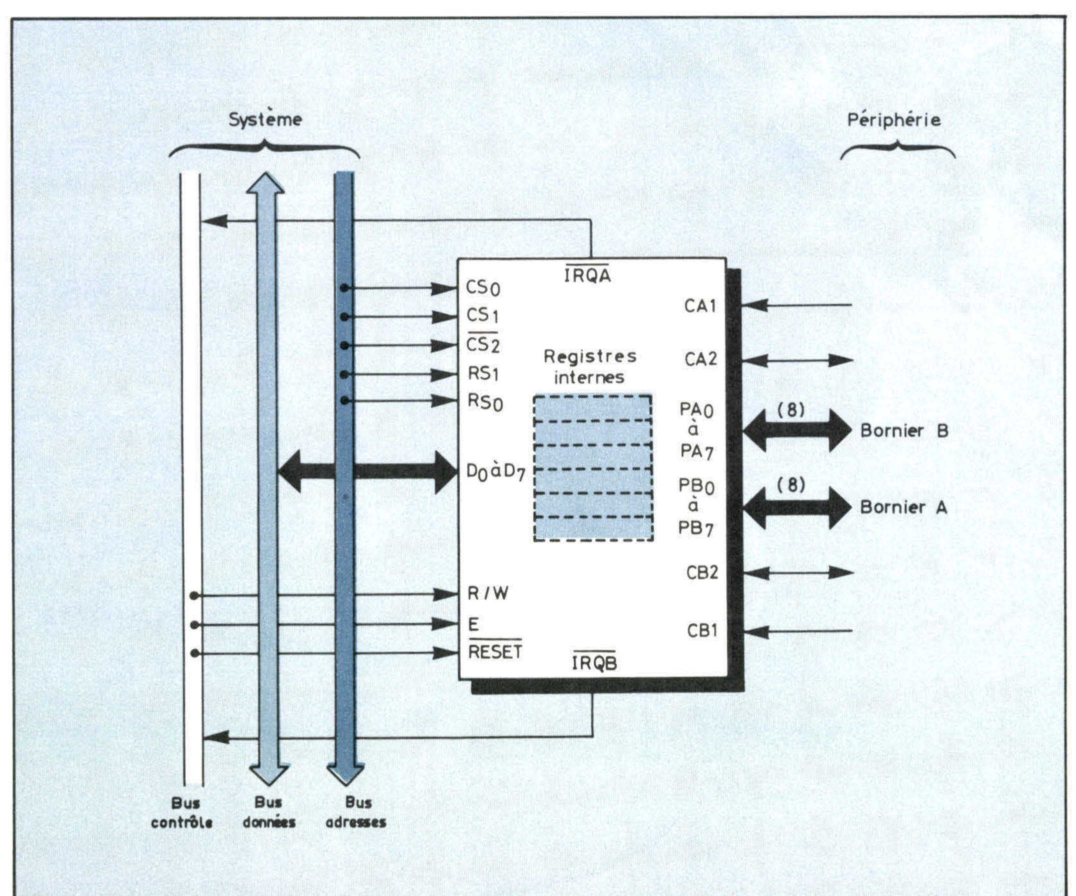



*Fig. 3. - Le P.I.A. communique entre le système et la périphérie à l'aide des signaux indiqués. Remarquez les borniers A ($PA_0$ à $PA_7$) et B ($PB_0$ à $PB_7$) ainsi que leurs lignes de dialogue associées ($CA_1$, $CA_2$, $\overline{IRQA}$) et ($CB_1$, $CB_2$, $\overline{IRQB}$).*

### Sélection et adressage des registres

Nous venons de voir que 6 registres internes peuvent être adressés par le microprocesseur qui les considère toutefois comme **4 adresses mémoire.**

Par 2 fils d'adressage, $RS_0$ et $RS_1$, on peut choisir parmi quatre registres. Un troisième fil aurait permis l'adressage de huit registres, alors qu'ils sont au nombre de six.

La solution choisie par le constructeur prévoit une économie de broches sur le boîtier. Ainsi, 2 registres sont adressés directement : **C R A et C R B** comme l'indique la **figure 5a ;** les quatre autres registres O R A, O R B et D D R A, D D R B étant adressés indirectement. Le choix entre O R A, O R B et D D R A, D D R B étant fonction du bit 2 écrit au préalable dans C R A, C R B **(fig. 5b et 5c).**

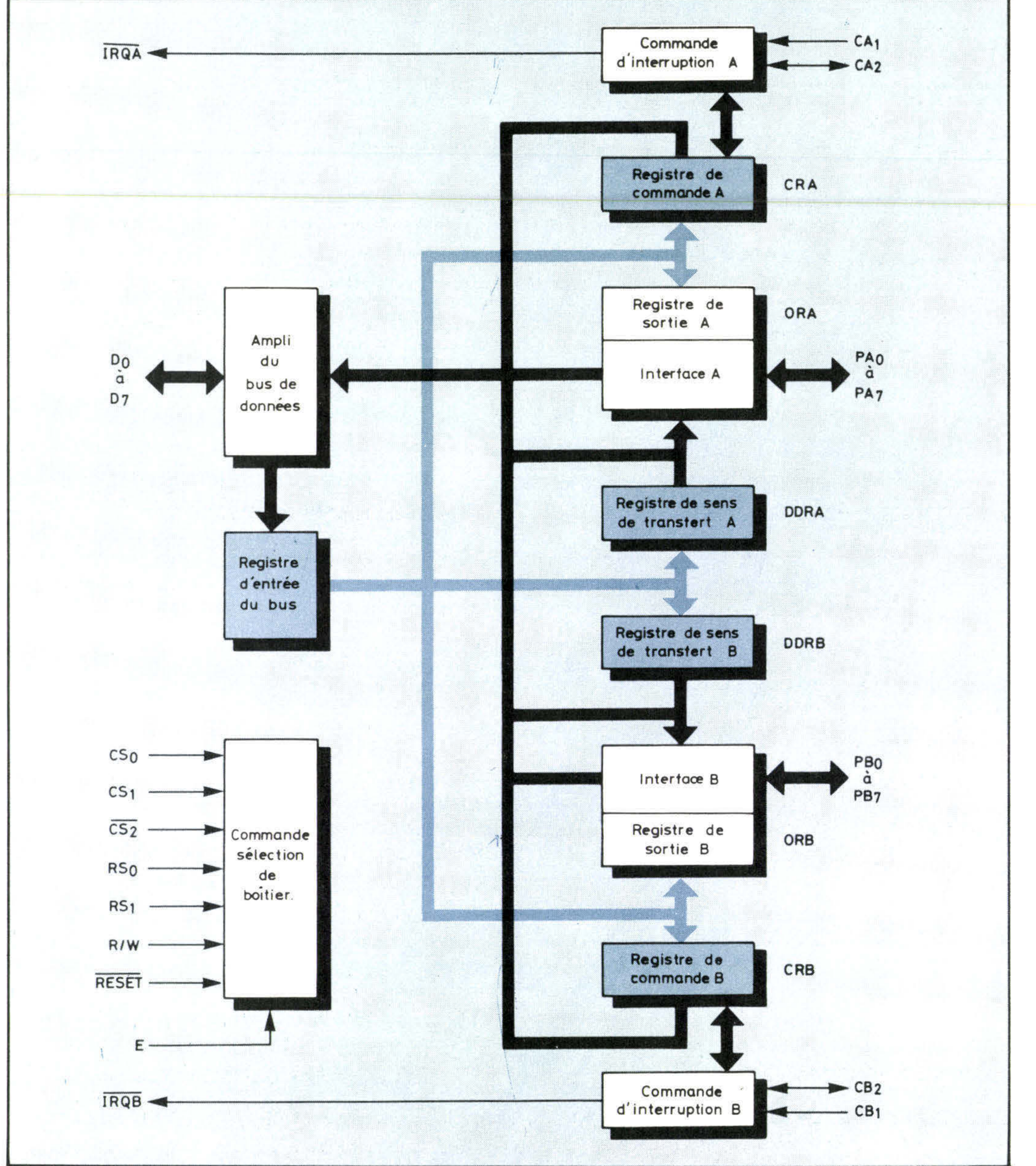
L'adressage des 6 registres du PIA peut donc se résumer sur le tableau correspondant suivant :

Les 6 registres internes du PIA sont adressés par le microprocesseur qui les considère comme 4 adresses mémoire.



| $RS_1$ | $RS_0$ | $CRA_2$ | $CRB_2$ | Registre adressé |
|---|---|---|---|---|
| 0 | 1 | - | - | C R A |
| 0 | 0 | 0 | - | D D R A |
| 0 | 0 | 1 | - | O R A et interface |
| 1 | 1 | - | -- | C R B |
| 1 | 0 | - | 0 | D D R B |
| 1 | 0 | - | 1 | O R B et interface |

*Fig. 4. - Schéma synoptique du MC 6820. Le microprocesseur peut adresser 6 registres en lecture et en écriture : CRA - DDRA - ORB ; CRB - DDRB - ORB.*




## Programmation

### Ecriture des registres « Sens de transfert » : D D R A et D D R B

Chacune des lignes des deux borniers $PA_0$-$PA_7$ et $PB_0$-$PB_7$ peut être, individuellement, programmée en entrée ou en sortie.

Ceci est obtenu par l'écriture du mot « sens de transfert » dans le registre D D R A ou D D R B.

Lorsqu'un « 0 » est écrit dans le bit i du registre D D R A, par exemple, le bit i de PA est programmé en entrée. Inversement, lorsqu'un « 1 » est écrit dans le bit i du registre D D R A, le bit i de PA est programmé en sortie.

Ainsi, dans le cas de la **figure 6** les bits 0, 4 et 6 du bornier $PA_0$-$PA_7$ sont des entrées et les bits 1, 2, 3, 5 et 7 des sorties.

Décrivons maintenant les transferts de données en tenant compte des différences technologiques des deux borniers A et B.

**Le bornier A : $PA_0$-$PA_7$**

Nous l'avons vu précédemment, il y a transfert de données sur les lignes programmées en sortie lorsque les bits correspondants du registre D D R A sont à « 1 ».

Dans l'exemple de la **figure 7a,** le microprocesseur écrit les bits $D_1$, $D_2$, $D_3$, $D_5$, $D_7$ à transmettre vers la périphérie dans le registre O R A, les autres bits n'étant pas pris en compte. Seuls ces bits apparaissent sur les lignes $PA_1$, $PA_2$, $PA_3$, $PA_5$, $PA_7$ et sont disponibles en permanence : **il y a mémorisation des sorties.**

Du point de vue électronique proprement dit, le schéma de fonctionnement est donné, à titre indicatif, sur la **figure 7b.** Lorsque D D R A est à 1, la porte ET autorise le transfert des informations de O R A vers PA.

Dans le cas où un ou plusieurs bits du registre D D R A est à 0, les lignes correspondantes sont programmées en entrées.

Les données présentes sur les lignes $PA_0$, $PA_4$ et $PA_6$ de la **figure 8a** apparaissent alors, après amplification, sur le bus de données du système afin que le microprocesseur puisse en effectuer la lecture ; toutefois, il n'y a pas mémorisation des entrées.

La **figure 8b** représente le schéma équivalent du circuit d'interface. Ici, $PA_6$ est transmis directement à $D_6$.

Il est à remarquer que lors de la lecture des lignes $PA_0$, $PA_4$, $PA_6$, le microprocesseur lit un mot de 8 bits dont seuls $D_0$, $D_4$, $D_6$ sont représentatifs.

Les autres bits lus représente-

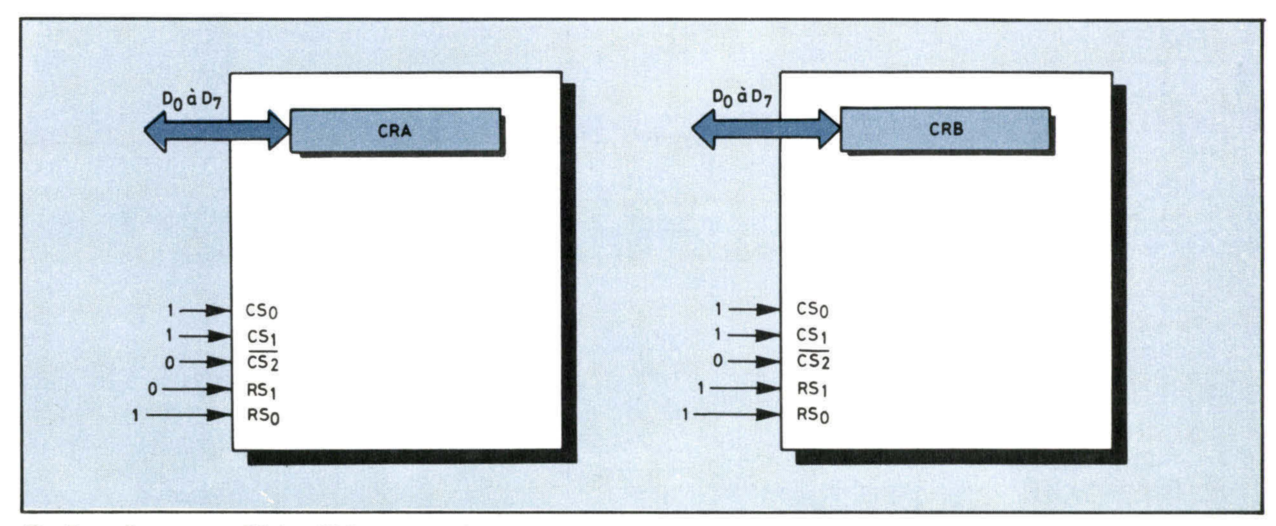



*Fig. 5 a. - Les registres CRA et CRB sont adressés directement grâce aux deux fils $RS_0$ et $RS_1$.*

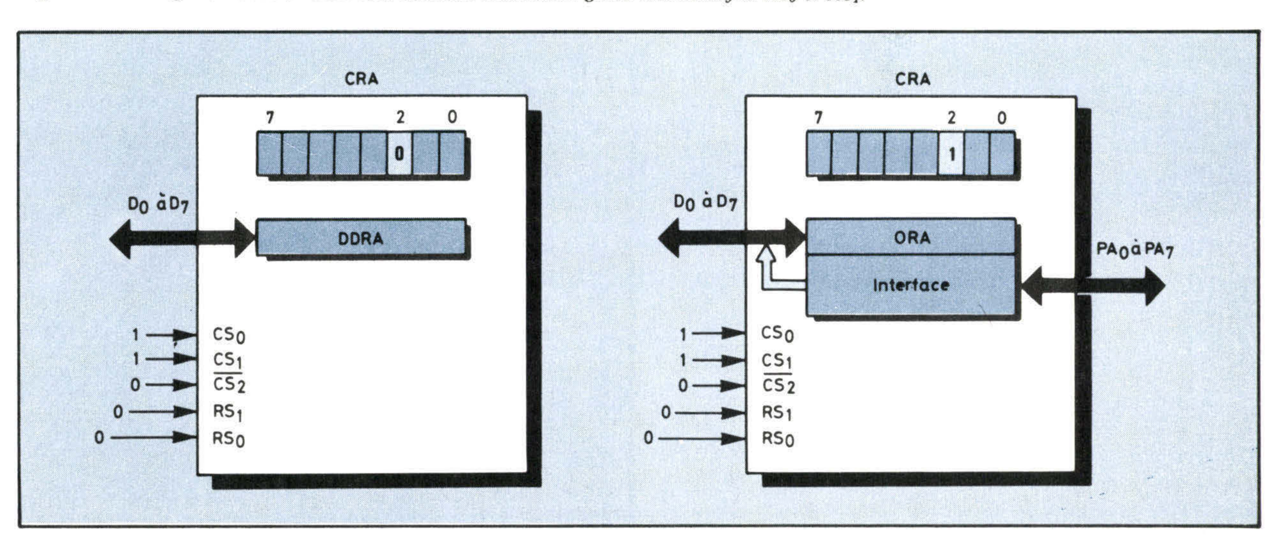



*Fig. 5 b. - Si $RS_0 = RS_1 = 0$, on adresse soit ORA soit DDRA en fonction du bit 2 écrit dans CRA. Si ce bit est à 0, le registre adressé est DDRA et s'il est à 1 le registre ORA est adressé.*

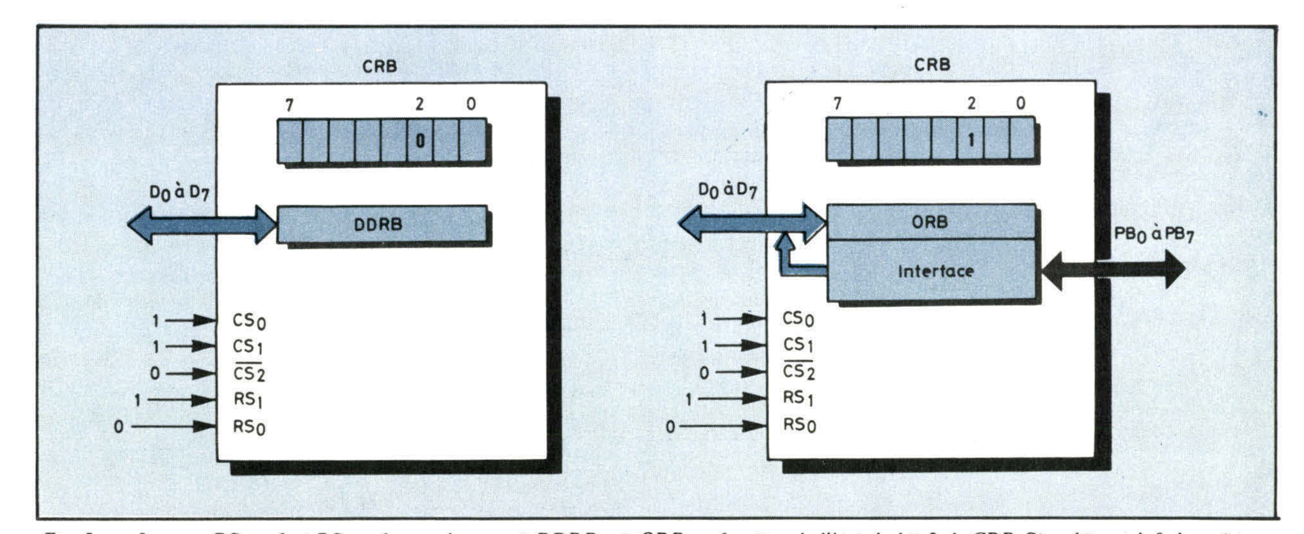



*Fig. 5 c. - Lorsque $RS_0 = 0$ et $RS_1 = 1$, on adresse soit DDRB soit ORB en fonction de l'état du bit 2 de CRB. Si ce bit est à 0, le registre adressé est alors DDRB ; s'il est à 1, c'est le registre ORB qui est adressé.*

ront l'état des sorties $PA_1$, $PA_2$, $PA_3$, $PA_5$, $PA_6$ ; si celles-ci sont chargées de façon à respecter les normes TTL, alors il y aura identité entre l'état de ces sorties et l'état des bits correspondants du registre O R A : tout se passe comme si le microprocesseur lisait l'état des bits du registre O R A pour les lignes programmées en sortie. Sinon il y aura une différence comme nous l'avons illustré dans l'exemple des **figures 9a et 9b.**

### Le bornier B : $PB_0$-$PB_7$

Chacune des lignes $PB_0$-$PB_7$ joue le même rôle que les lignes $PA_0$-$PA_7$ du bornier A. Il y a simplement une différence de structure, les lignes PB pouvant être mises à l'état haute impédance (tristate) lorsqu'elles sont programmées en entrée.

La **figure 10** représente le schéma électronique du système lors d'un transfert de données sur les lignes programmées en sortie. Puisqu'il s'agit d'une sortie, les données présentes sur le bus de données sont mémorisées dans le registre O R B, le bit 4 à 1 du registre D D R B définit le sens du transfert des informations : registre O R B vers sortie $PB_4$.

On vérifie aisément que lorsque le bit 4 de O R B est à « 0 », $PB_4$ est à « 0 » et lorsqu'il est à 1, $PB_4$ passe à l'état « 1 ».

Dans le cas où les lignes sont programmées en entrées, les bits correspondants de D D R B sont à 0 et l'état de ces lignes de sortie est transmis sur le bus de données.

L'exemple de la **figure 11** illustre la transmission de l'information présente sur $PB_3$ vers $D_3$ puisque le bit 3 de D D R B est à 0.

Il est à noter qu'une **lecture par l'unité centrale des lignes programmées en sortie** entraîne la lecture des bits correspondants du registre O R B, indépendamment de l'état des lignes de sortie.

### Ecriture des registres de commande

Nous avons vu précédemment que parmi les 6 registres internes au PIA, 2 d'entre eux C R A et

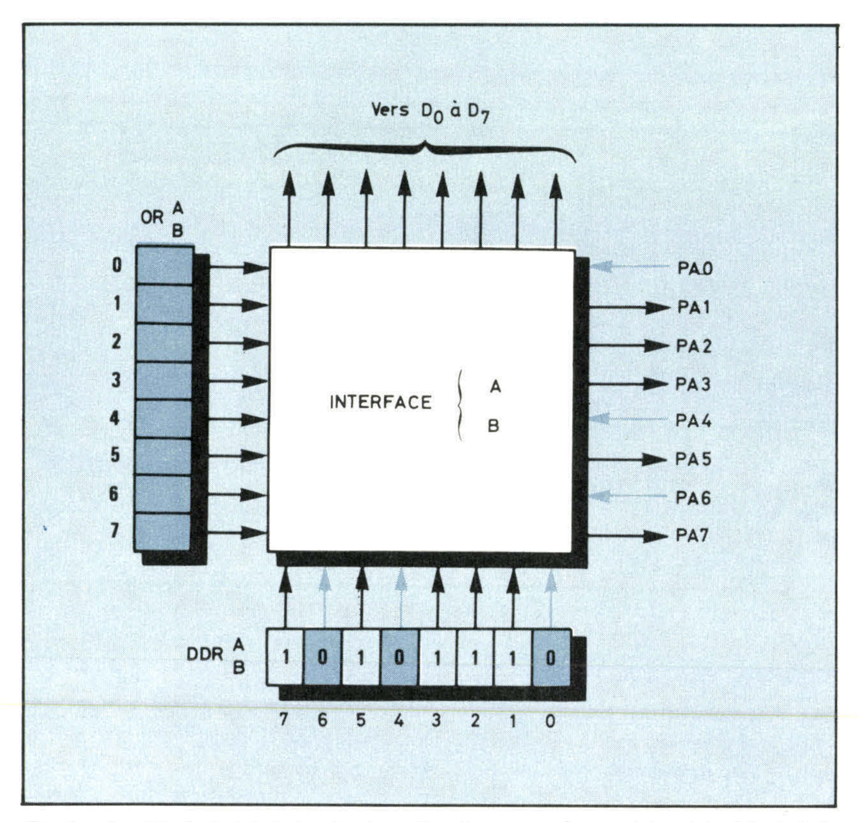



*Fig. 6. – Les bits 0, 4 et 6 du bornier A ou B sont programmés en entrées et les bits 1, 2, 3, 5, 7 sont programmés en sorties en fonction de l'état des bits présents dans le registre DDRA ou DDRB.*

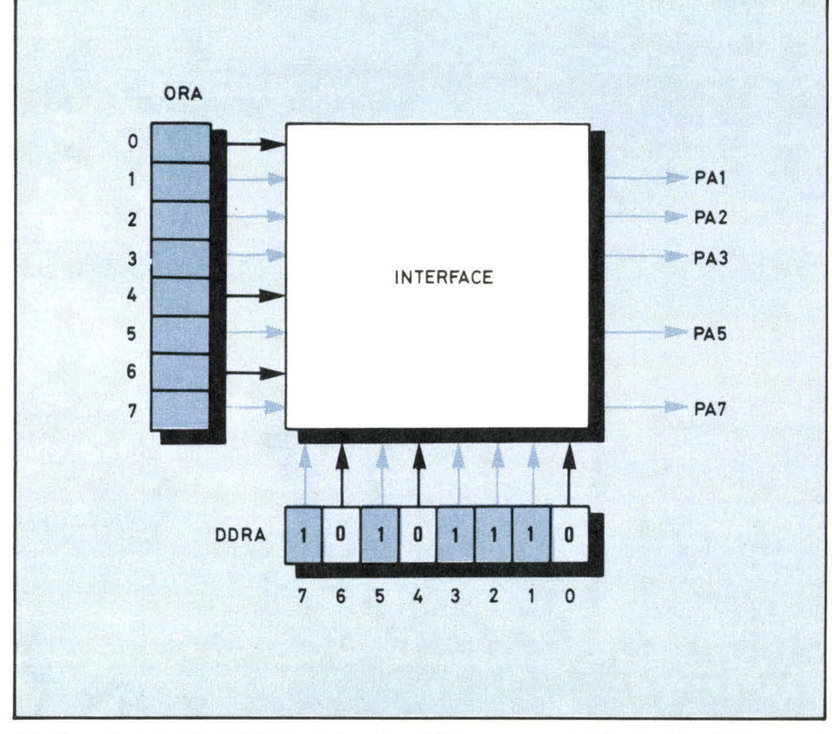



*Fig. 7 a. – Seuls les bits 1, 2, 3, 5 et 7 du registre ORA sont transmis à la périphérie (DDRA = 1) sur les lignes $PA_1$, $PA_2$, $PA_3$, $PA_5$ et $PA_7$ où ils restent disponibles en permanence car ils sont mémorisés en ORA.*

*Fig. 7 b. – Les bits à 1 du registre DDRA commandent une porte ET et permettent le transfert des bits choisis du registre ORA vers les lignes $PA_i$.*

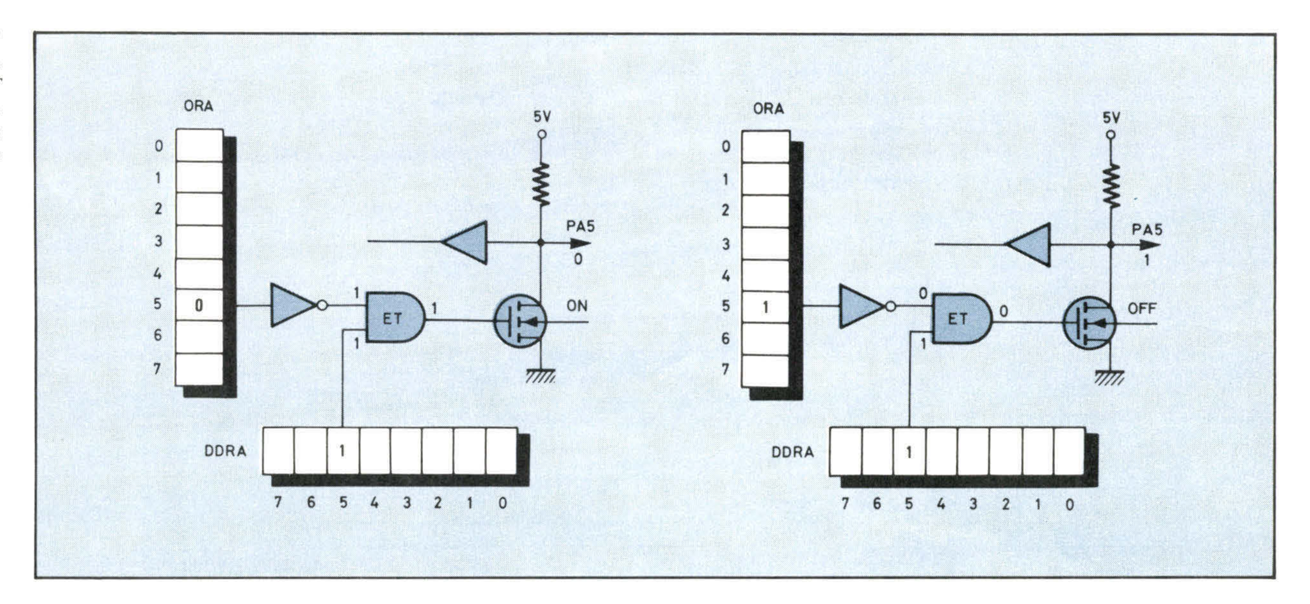



*Fig. 8 a. – Les bits à 0 de DDRA permettent de programmer en entrées les bits correspondants des lignes $PA_i$.*

*Fig. 8 b. – Les bits à 0 du registre DDRA sont appliqués à une porte ET. Sa sortie à niveau bas bloque le transistor. Les lignes issues de la périphérie et programmées en entrées sont alors transmises vers le bus de données du système. Ici $PA_6$ est transmise directement à $D_6$.*

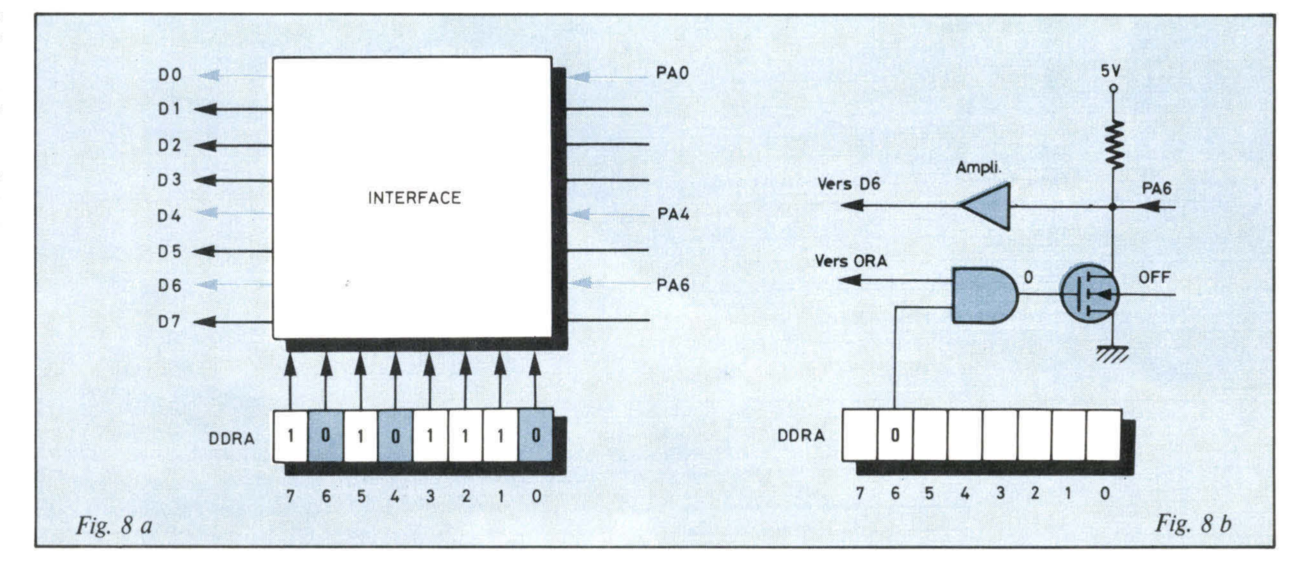



Fig. 8 a

Fig. 8 b

*Fig. 9 a. – Le microprocesseur lit l'état de $PA_5$ qui correspond bien à celui du bit 5 de ORA lorsque les sorties sont chargées de manière à respecter les normes TTL.*

*Fig. 9 b. – Le microprocesseur lit l'état de $PA_5$, différant de celui du bit 5 de ORA en raison de la charge faible.*

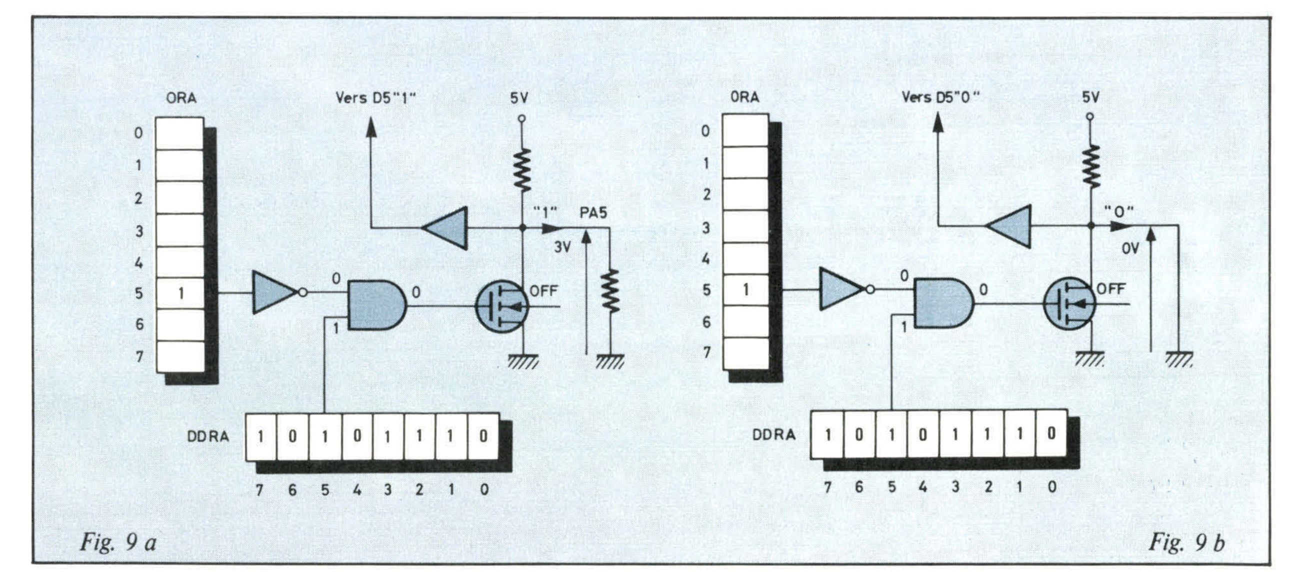



Fig. 9 a

Fig. 9 b

Chacune des lignes des 2 ports peut être, individuellement, programmée en entrée ou en sortie.



C R B contiennent les paramètres de fonctionnement du circuit.

Dans ces 2 registres, les bits 2 permettent de définir l'adressage des autres registres. Tous les autres bits de C R A et C R B étant relatifs aux lignes d'interruptions $CA_1$, $CB_1$ ; $CA_2$, $CB_2$ ; $\overline{IRQA}$ et $\overline{IRQB}$ disponibles sur le boîtier 40 broches.

Le format des registres C R A et C R B est donné **figure 12.** Les bits 6 et 7 de ces 2 registres ne peuvent être écrits et sont seulement lus.

**Mode de fonctionnement de $CA_1$**

La programmation du mode de fonctionnement de la ligne d'interruption $CA_1$ est décrite dans le tableau ci-dessous :

| N° | $CRA_1$ | $CRA_0$ | transition active de l'entrée d'int. $CA_1$ | indicateur d'interruption $CRA_7$ | sortie d'interruption $\overline{IRQA}$ (vers MPU) |
|---|---|---|---|---|---|
| 1 | 0 | 0 | | mis à 1 par | interruption $\overline{IRQA}$ = 1 masquée |
| 2 | 0 | 1 | | mis à 1 par | passe à 0 quand $CRA_7$ passe à 1 |
| 3 | 1 | 0 | | mis à 1 par | interruption $\overline{IRQA}$ = 1 masquée |
| 4 | 1 | 1 | | mis à 1 par | passe à 0 quand $CRA_7$ passe à 1 |

Par exemple, si nous prenons la première ligne de ce tableau, lorsque les bits 1 et 0 de C R A sont à « 0 », une demande d'interruption est prise en compte sur un front descendant de $CA_1$, l'indicateur d'interruption $CRA_7$ associé à $CA_1$, est mis à 1 et le PIA génère un signal de sortie d'interruption $\overline{IRQA}$ vers le microprocesseur.

L'indicateur d'interruption $CRA_7$ **est mis à zéro** par une lecture du registre de données de la périphérie A par le microprocesseur.

Une demande d'interruption masquée par $CRA_0$ = « 0 » est cependant mémorisée et devient active lorsque $CRA_0$ passe à « 1 » ; $\overline{IRQA}$ est activée à 0.

L'illustration graphique des modes de fonctionnement des lignes d'interruption est donnée **figure 13.**

*Fig. 10. – Schéma électronique du système lors d'un transfert de données sur les lignes programmées en sorties. Lorsque le bit 4 de DDRB est à 1 le transfert sur la ligne $PB_4$ de la donnée contenue dans le bit 4 de ORB est autorisé.*

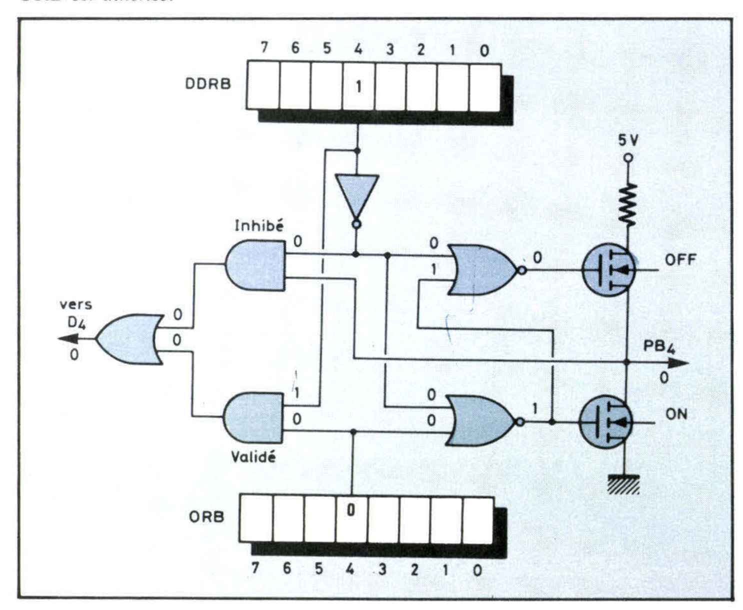
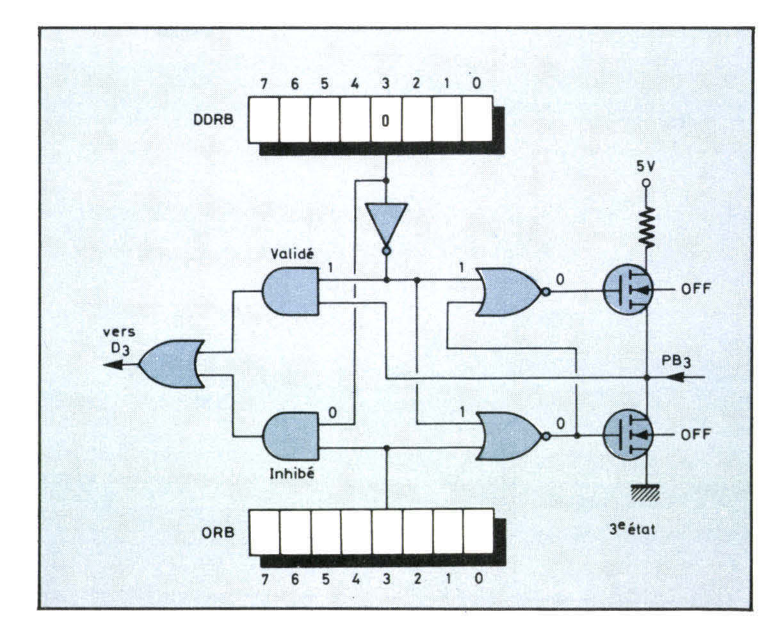



*Fig. 11. – Schéma électronique du système lors d'un transfert de données des lignes programmées en entrées. Le bit n° 3 de DDRB étant à 0, le contenu du bit 3 de ORB est transmis vers $D_3$.*



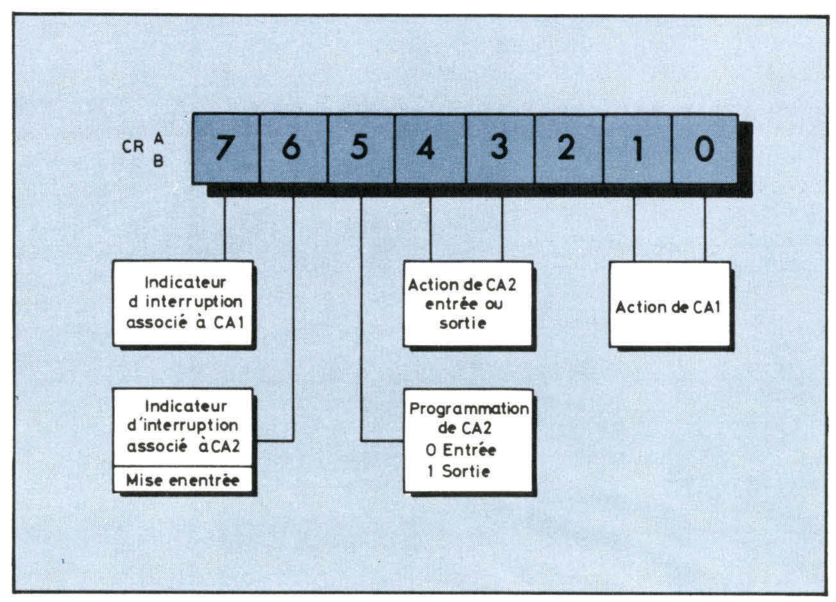



*Fig. 12. – Format des registres de commande CRA et CRB. Les bits 6 et 7 ne peuvent pas être écrits et sont uniquement lus.*

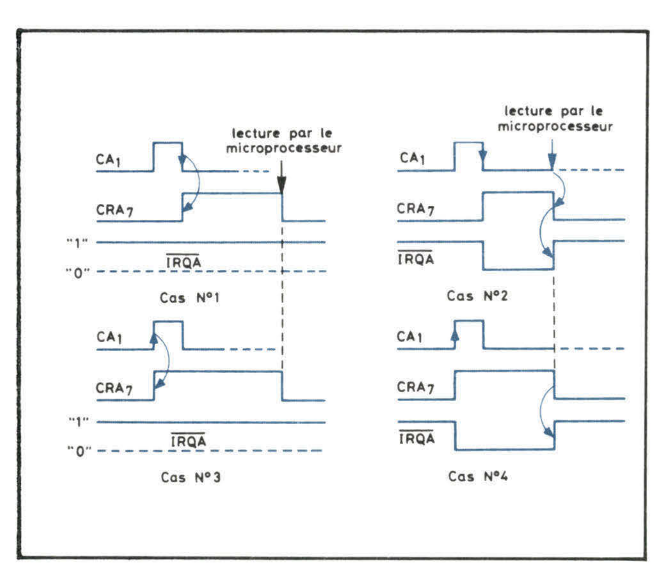



*Fig. 13. – Modes de fonctionnement des lignes d'interruptions.*

## Mode de fonctionnement de $CA_2$

La programmation du mode de fonctionnement de la ligne d'interruption $CA_2$ mise en entrée est identique à $CA_1$ si le bit 5 du mot de commande C R A est à 0.

Dans ce cas, les bits $CRA_3$, $CRA_4$ et $CRA_6$ jouent le même rôle que les bits $CRA_0$, $CRA_1$ et $CRA_7$.

La programmation de $CA_2$ en sortie de commande s'obtient en écrivant un « 1 » dans $CRA_5$. $CRA_4$ et $CRA_3$ permettent de définir les modes d'action de $CA_2$ **(fig. 14).**

Selon la programmation des bits $CRA_4$ et $CRA_3$ on distingue 4 modes de fonctionnement :

| $CRA_4$ | $CRA_3$ | Modes |
|---|---|---|
| 0 | 0 | dialogue } associés à une lecture |
| 0 | 1 | impulsionnel } associés à une lecture |
| 1 | 0 | programmé |
| 1 | 1 | programmé |

Dans le **mode programmé,** la sortie $CA_2$ suit la programmation du bit $CRA_3$ du registre CRA.

Dans le cas des modes impulsionnel et dialogue, $CA_2$ est associé à une **lecture.**

Sur la **figure 15,** nous remarquons que $CA_2$ est activé au niveau bas par le front descendant de l'impulsion E qui active une lecture des données en provenance du bornier A, ceci quel que soit le mode de fonctionnement.

**Dans le mode impulsionnel,** $CA_2$ est remis au niveau haut par le front descendant de la première impulsion E qui suit une désélection du circuit.

Remarquons que dans le cas d'un fonctionnement normal, après lecture du bornier A le microprocesseur exécute une instruction dont il vient de faire l'acquisition : le PIA est donc désélecté, par suite $CA_2$ a une durée d'un cycle d'horloge.

**Dans le mode dialogue** $CA_2$ est remis au niveau haut par l'indicateur d'interruption $CRA_7$, lui-même mis à un par le front actif de $CA_1$, ceci de façon **asynchrone.**

## Mode de fonctionnement de $CB_2$

De même que pour $CA_2$, la programmation de cette ligne en sortie de commande s'obtient en écrivant $CRB_5 = 1$.

Mais $CRB_4$ et $CRB_3$ permettent de définir des modes d'action de $CB_2$ **différents** de $CA_2$.

| $CRB_4$ | $CRB_3$ | Modes |
|---|---|---|
| 0 | 0 | dialogue } associés à une écriture |
| 0 | 1 | impulsionnel } associés à une écriture |
| 1 | 0 | programmé |
| 1 | 1 | programmé |

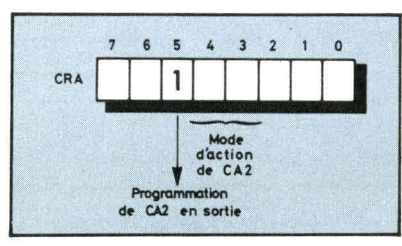



*Fig. 14. – L'écriture d'un 1 dans $CRA_5$ permet la programmation en sortie de commande de $CA_2$. Les modes d'action de $CA_2$ sont alors définis par $CRA_3$ et $CRA_4$.*

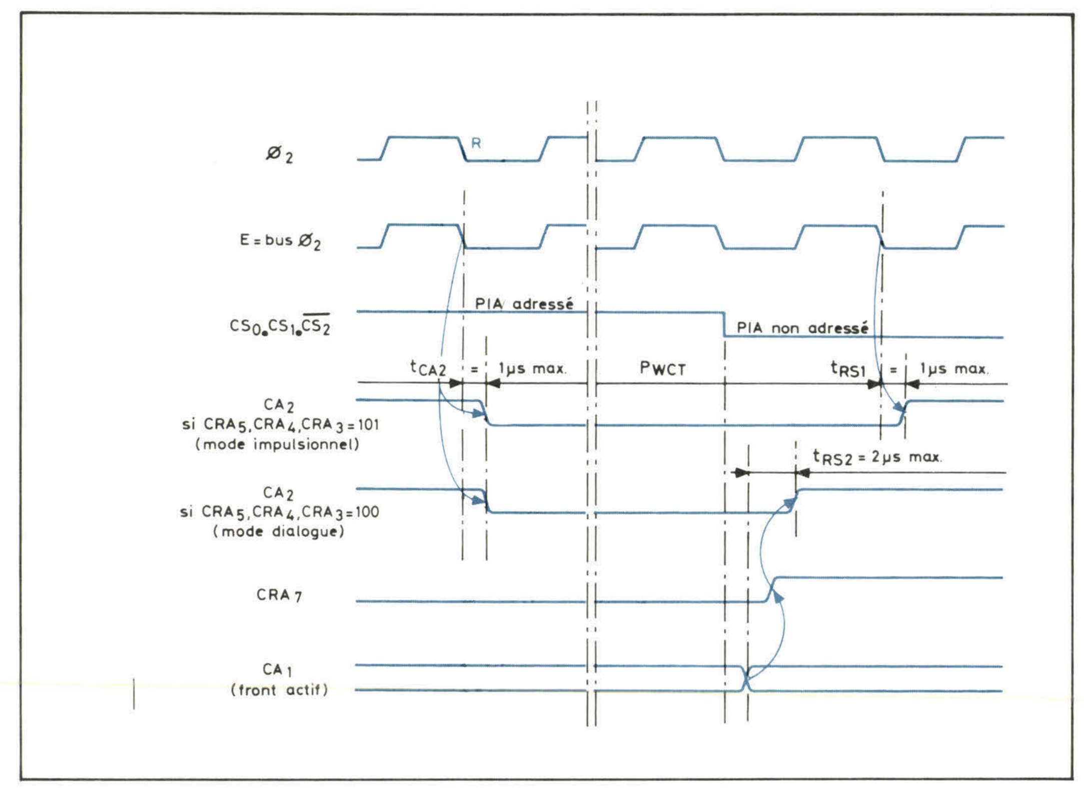
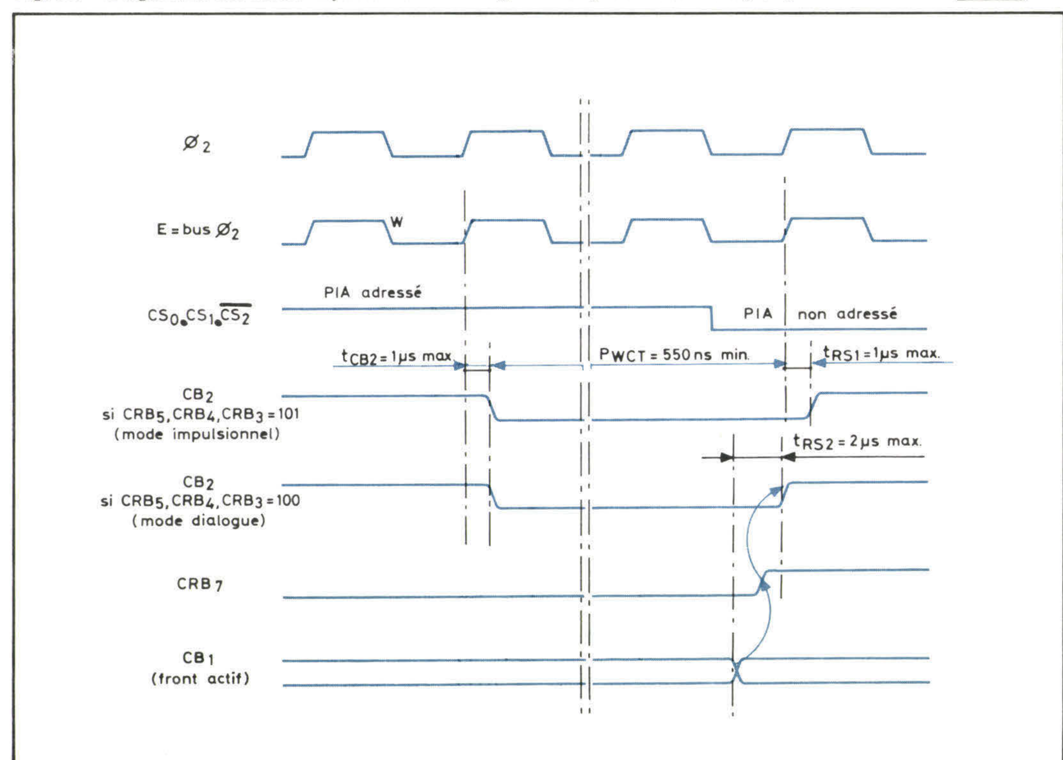



*Fig. 15. – Diagramme des modes impulsionnels et dialogue de $CA_2$. Pour ces modes, $CA_2$ est associé à une lecture.*



*Fig. 16. – Diagramme des modes impulsionnels et dialogue de $CB_2$. Pour ces modes, $CB_2$ est associé à une écriture.*




Dans le mode programmé, la sortie $CB_2$ suit la programmation du bit $CRB_3$ du registre CRB.

Contrairement à $CA_2$, dans les modes impulsionnel et dialogue, $CB_2$ est associé à une **écriture.**

Ainsi, sur le schéma de la **figure 16,** $CB_2$ est activé au niveau bas par le front montant de la première impulsion E qui suit une écriture des données par le microprocesseur, ceci quel que soit le mode de fonctionnement.

Dans le **mode impulsionnel** $CB_2$ est remis au niveau haut par le front montant de la première impulsion E qui suit une désélection du circuit.

Dans le cas d'un fonctionnement normal, voir l'explication donnée à propos de $CA_2$, $CB_2$ à une durée d'un cycle d'horloge.

Dans le **mode dialogue** $CB_2$ est remis au niveau haut par l'indicateur d'interruption $CRB_7$, lui-même mis à un par le front actif de $CB_1$ et ceci de façon **asynchrone.**

## $\overline{\text{RESET}}$

La ligne $\overline{\text{RESET}}$, active au niveau bas, permet de remettre les registres internes du PIA à zéro.

Il est recommandé de maintenir à « 1 » les lignes $CA_1$, $CA_2$, $CB_1$, $CB_2$ quand $\overline{\text{RESET}}$ est actif, afin d'éviter la mise à « 1 » des indicateurs d'interruption correspondants sur la transition positive de $\overline{\text{RESET}}$.

Une lecture des registres données peut être cependant faite pour remettre à zéro ces indicateurs intempestivement activés.

Dans notre prochain numéro nous publierons un exemple d'application, qui se veut exhaustif, de l'utilisation d'un PIA. En effet, nous traiterons le cas d'un multiplieur-diviseur 8 bits par 8 bits, microprogrammé, appelé MULDIV. ■

**P. PELLOSO ***
*(ING. CNAM)*
**A. SEMETEYS ***
*(ING. EFR)*

** P. Pelloso et A. Semeteys sont enseignants à l'IUT de Créteil et en formation continue au centre Ouest Parisien associé au CNAM (Gestop). Ils sont aussi chargés d'études auprès de l'E.D.F.*

# Les interruptions

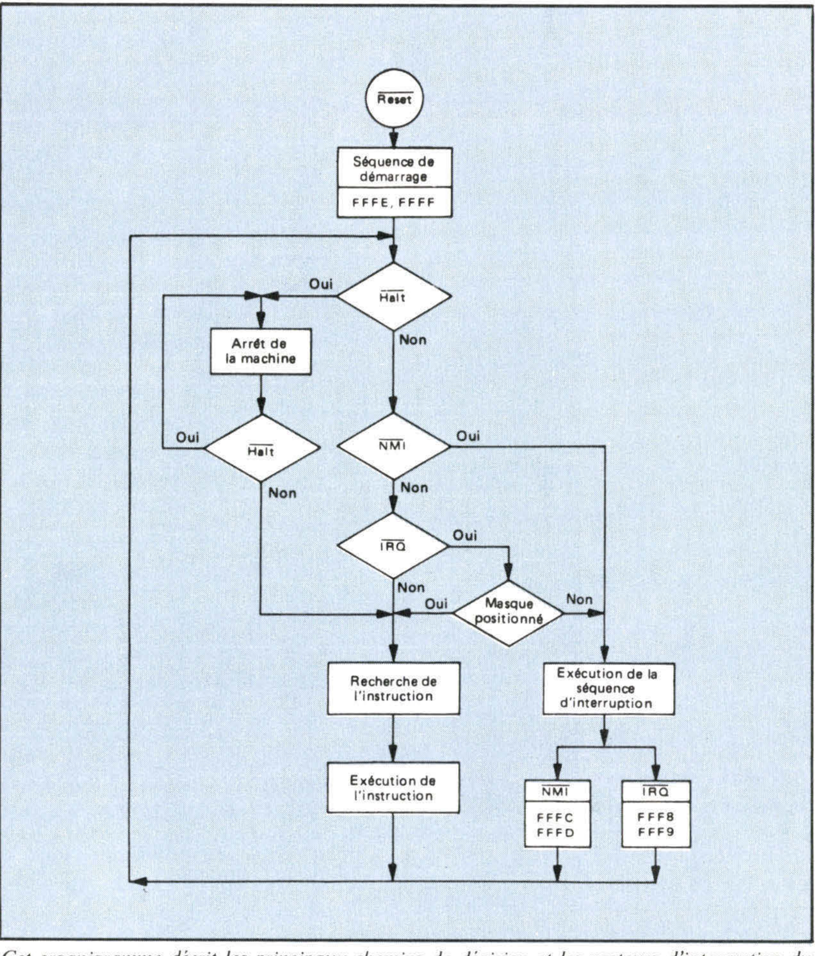
**Pour exécuter efficacement certaines tâches et faciliter le travail de l'utilisateur, un microprocesseur doit offrir, entre autres, la possibilité de gérer aisément des interruptions.**

**Les interruptions sont des événements qui provoquent l'arrêt d'un programme en cours de traitement et le passage à un autre programme.**

**Les domaines d'utilisation d'un processus d'interruption sont assez vastes, nous retiendrons par exemple les interruptions dues aux erreurs ou aux pannes machine (panne d'alimentation, erreur de parité en mémoire...) ; les interruptions de programme (instruction ou adresse incorrecte, opération impossible, division par zéro...) ; les interruptions d'entrée-sortie (besoin d'échange d'informations entre le microprocesseur et les périphériques : téléimprimeur, clavier, écran de visualisation, capteurs...).**

**L'usage des interruptions donne ainsi aux concepteurs de systèmes une souplesse d'emploi considérable.**

**Dans cet article, nous traiterons successivement des notions de pile et de pointeur de pile, puis nous aborderons les différents types d'interruptions telles que les interruptions prioritaires, les interruptions masquables, les interruptions logicielles, l'initialisation et les interruptions vectorisées. Enfin, nous analyserons le principe des priorités dans les interruptions.**




*Cet organigramme décrit les principaux chemins de décision et les vecteurs d'interruption du microprocesseur 6800. A l'initialisation ($\overline{RESET}$), le compteur de programme est chargé avec l'adresse FFFE, FFFF et la séquence de démarrage s'effectue. Si la broche $\overline{HALT}$ est à un niveau bas (oui), le microprocesseur est arrêté et se maintient dans cet état jusqu'au moment où $\overline{HALT}$ passe à un niveau haut, ce qui déclenche la recherche et l'exécution de l'instruction. Si $\overline{HALT} = 1$ (non) et qu'une interruption non masquable ($\overline{NMI}$) est demandée, le vecteur correspondant contenu en FFFC, FFFD est lu par le MPU et le sous-programme d'interruption non masquable s'exécute. Dans le cas où une interruption non masquable (prioritaire) n'a pas été demandée et que survient une interruption masquable, celle-ci se déroule (à condition que le masque soit positionné ($I = 0$)) grâce à la vectorisation en FFF8 et FFF9. Si le masque n'a pas été positionné ou s'il n'y a pas eu de demande d'interruption, il y a directement recherche et exécution de l'instruction.*

## Notions de « pile »

Pour pouvoir accepter une interruption, le microprocesseur doit suspendre l'exécution du programme qu'il est en train de traiter. Pour cela, il doit stocker les informations contenues dans l'ensemble des registres internes, puisque ce sont eux qui contrôlent le bon déroulement du programme en association avec les instructions stockées en mémoire.

Ainsi, à la réception d'une demande d'interruption, le contenu des registres internes du microprocesseur est préservé dans des emplacements mémoires appelés **PILE** (Stack en anglo-saxon).

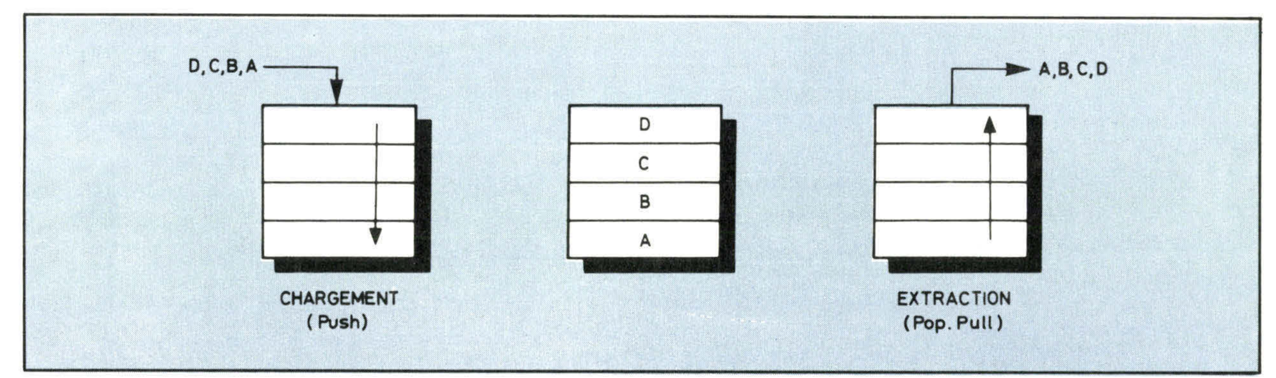



*Fig. 1. – Exemple de pile LIFO. La dernière donnée mémorisée (D) lors du chargement est la première restituée lors de l'extraction.*

Une pile est une zone mémoire utilisée pour stocker temporairement les informations contenues dans l'ensemble des registres internes du microprocesseur.

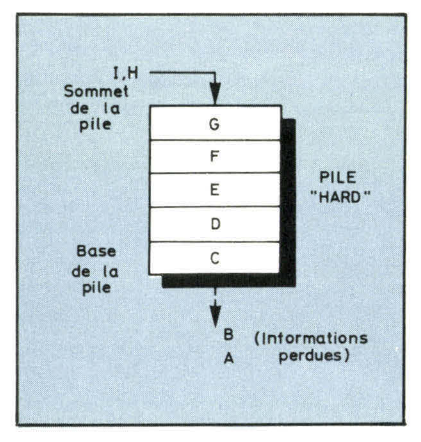



*Fig. 2. – Lorsqu'une pile « hard » est chargée au-delà de sa capacité les premières informations rentrées sont perdues, il y a « écrasement » des données.*

Après l'exécution du programme d'interruption, il y aura restitution du contenu des registres, c'est-à-dire transfert de données des emplacements mémoire vers les registres internes.

## Structure d'une pile

Une pile est donc une zone mémoire utilisée pour stocker temporairement une ou plusieurs informations. Elle est quelquefois appelée LIFO (Last In, First Out). Cette appellation est justifiée par le fait que le dernier événement stocké dans la pile sera le premier ressorti.

Un exemple de pile LIFO de quatre emplacements mémoire est illustré **figure 1.** Les données issues des registres internes sont chargées successivement dans l'ordre A, B, C et D. Lors de la restitution, l'extraction de ces valeurs s'effectue dans un ordre inversé : D, C, B et A.

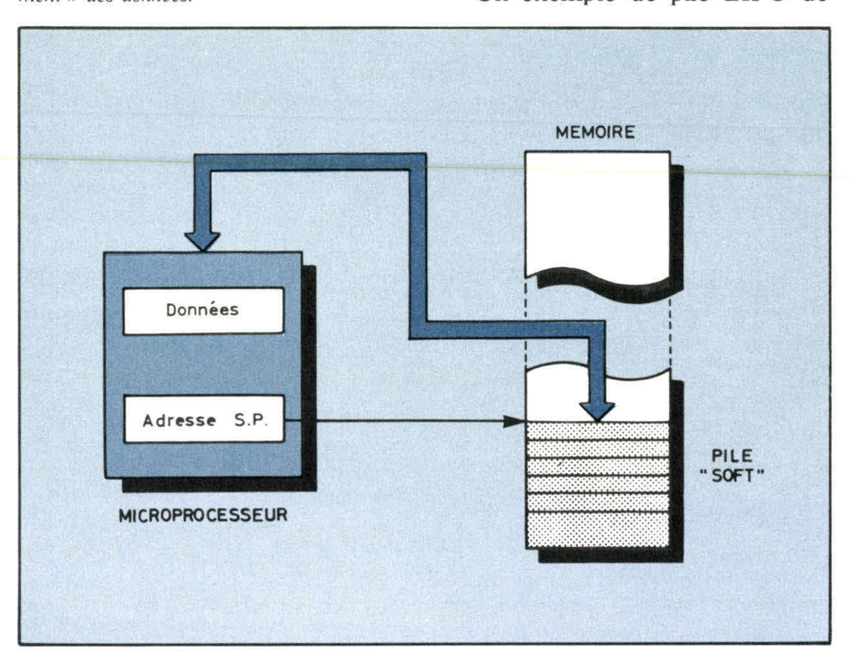
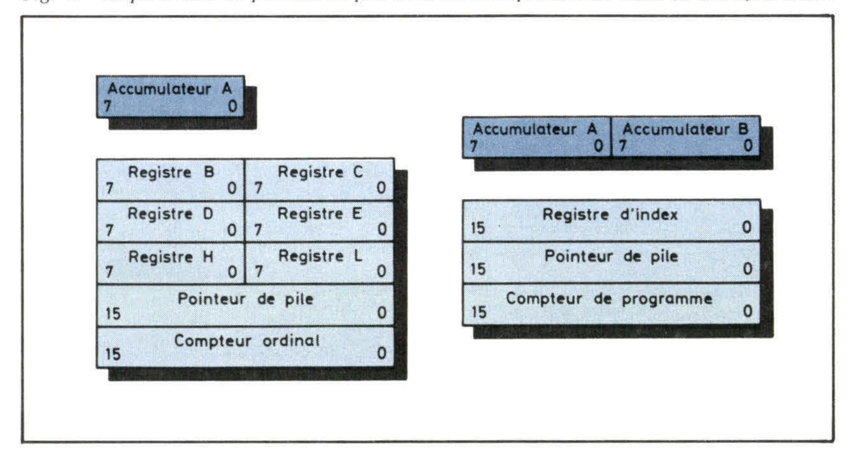



*Fig. 3. – Afin de permettre un empilement rapide des informations le registre S.P. pointe toujours le mot situé juste au-dessus du dernier mot déposé.*

*Fig. 4. – Emplacement du pointeur de pile dans les microprocesseurs 6800 (à droite) et 8080.*




## Réalisation d'une pile

Deux techniques sont utilisées pour réaliser une pile : la pile en « matériel » ou la pile en « logiciel ».

● **La pile en matériel**

Une pile en matériel ou « hard » est constituée par un ensemble de registres internes au microprocesseur.

L'avantage est bien sûr une grande rapidité dans l'utilisation de ces registres, le désavantage est leur capacité limitée. Si la pile est chargée au-delà de sa capacité, le contenu du dernier mot dans la pile est perdu chaque fois qu'un mot supplémentaire est poussé au sommet de la pile. Dans l'exemple de la **figure 2** les premières données A et B introduites dans la pile « hard » sont perdues ou « écrasées ».

● **La pile dite programmée**

Ce type de pile est souvent appelé « soft ». Elle consiste simplement en une zone réservée dans la mémoire vive du système.

La position de cette pile dans la mémoire est choisie par le programmeur. Pour définir l'adresse de la pile, il existe à l'intérieur du microprocesseur un compteur spécial appelé pointeur de pile (Stack Pointer : **SP).**

Le registre S.P. pointe toujours, au sommet de la pile, le mot situé juste au-dessus du dernier mot déposé. Ceci a pour but de permettre un empilement plus rapide des informations **(fig. 3).**

A titre d'exemple, la **figure 4** indique l'emplacement du pointeur de pile parmi les registres des microprocesseurs 6800 et 8080. Il est intéressant de noter que le S.P. a une longueur de 16 bits puisqu'il définit une adresse parmi 64 k.

Dans tout programme utilisant la pile, ou prévoyant une interruption, il faut commencer par initia-

liser le pointeur de pile, c'est-à-dire indiquer **l'adresse du sommet de la pile.**

Lors d'une opération de chargement (pushing), il y a transfert du contenu de l'accumulateur à l'adresse indiquée par le stack pointer, puis **décrémentation** de celui-ci (il se déplace vers les adresses décroissantes).

Lors d'une opération d'extraction (popping) il y a **incrémentation** du stack pointer **puis** transfert du contenu de la position mémoire repérée par le pointeur de pile, dans l'accumulateur.

### Sauvegarde lors d'une interruption

Lors d'une demande d'interruption, le contenu des registres internes du microprocesseur est sauvegardé automatiquement dans la pile.

Lorsque la cause d'interruption a été exécutée, tous ces registres sont désempilés, puis réinstallés à l'intérieur du microprocesseur et le programme qui avait été suspendu pourra reprendre son exécution.

L'ordre dans lequel sont empilées les informations varie d'un microprocesseur à l'autre, nous indiquons ici à titre d'exemple, la sauvegarde du 6800 **(fig. 5).**

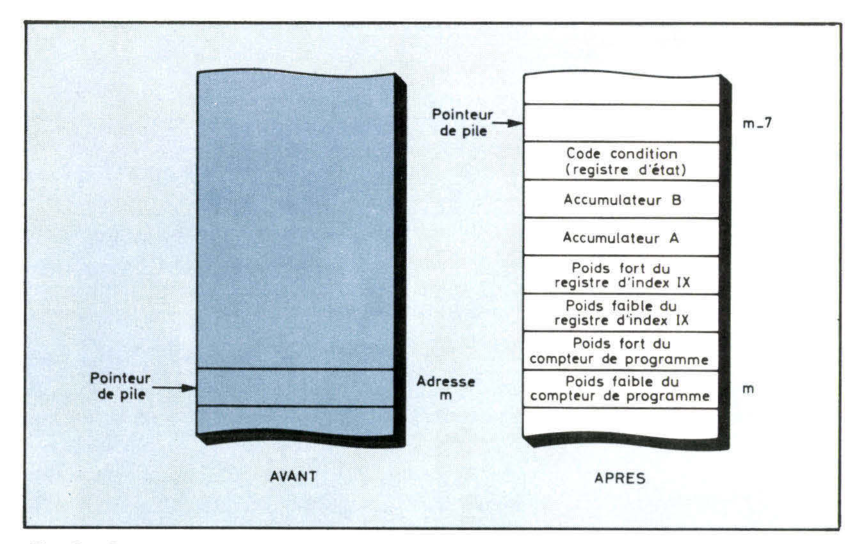



*Fig. 5. – Sauvegarde du contenu des différents registes internes du 6800 dans la pile lors d'une interruption.*

## Les différents types d'interruptions

### Interruption prioritaire

Il existe dans certains microprocesseurs, une entrée d'interruption prioritaire. Dès qu'une demande d'interruption est faite sur cette borne, l'unité de commande termine d'exécuter son instruction en cours, puis automatiquement transfère dans la pile les informations à sauvegarder.

Il s'agit par exemple de la borne NMI (**N**on **M**askable **I**nterrupt) dans le 6800. Dans ce cas, sur un front descendant (passage du niveau haut au niveau bas sur cette borne) l'interruption a lieu systématiquement.

### Interruption masquable par programme

Tous les microprocesseurs possèdent une entrée d'interruption masquable. Cette **interruption n'aura lieu** que si une bascule interne au microprocesseur, positionnable par des instructions, est **à l'état 0.**

Cette bascule porte le nom de masque d'interruption et peut se trouver dans le registre d'état (par exemple condition I dans le 6800) ou indépendante de ce registre (bascule INTE dans le 8080).

Dans le 8080, l'instruction EI (Enable Interrupt) autorise l'interruption en plaçant un « 1 » dans la bascule INTE qui indique que les interruptions décelées doivent être prises en considération.

L'instruction DI (Disable Interrupt), quant à elle, interdit l'interruption en mettant un 0 dans la bascule INTE.

Si nous prenons l'exemple du 6800, la borne d'interruption masquable s'appelle **IRQ** (Interrupt Request) et elle est active au niveau bas, l'interruption ayant lieu lorsque I = 0.

La condition I est positionnée par deux instructions :
Clear Interrupt Mask (CLI) : I est positionnée à 0.
Set Interrupt Mask (SEI) : I est positionnée à 1.

Il est à noter qu'après chaque interruption, le bit I est automatiquement mis à 1. Il ne faudra donc pas oublier de le remettre à zéro, lors de l'exécution du programme.

### Interruption logicielle

Parmi les instructions dont sont dotés les microprocesseurs, certaines d'entre elles provoquent une interruption. Ces interruptions par programme sont, en général, non masquables.

Il s'agit, par exemple, de l'instruction SWI (Software Interrupt) dans le 6800 ou du RESTART (RST) dans le 8080 de Intel.

RESTART est une instruction spéciale de branchement de sous-programme. Elle est codée sur 8 bits suivant le format indiqué **figure 6.**

Lors de l'exécution de cette instruction, le contenu du compteur ordinal est transféré dans la pile et représente l'adresse de retour, le

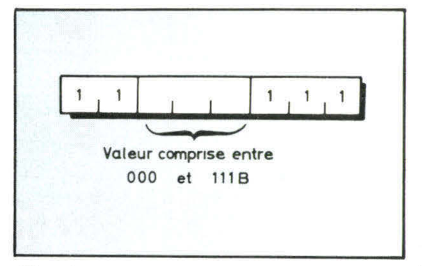



*Fig. 6*

Une interruption est vectorisée lorsqu'elle provoque un branchement direct à l'adresse de son programme de traitement.

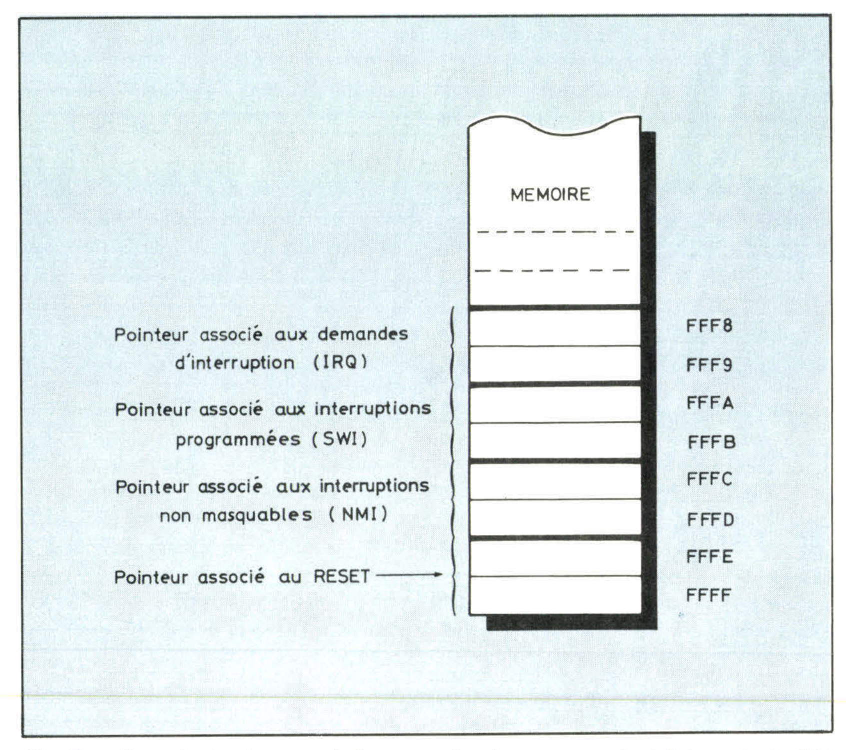



*Fig. 7. – Organisation interne de la zone mémoire correspondant à la vectorisation des interruptions dans un 6800.*

programme se poursuit alors à l'adresse indiquée par le mot de 3 bits compris entre 000 et 111 B.

RST est utilisée en liaison avec huit sous-programmes au maximum, chacun d'entre eux pouvant avoir une longueur de 8 octets. Ainsi, les 64 mots inférieurs de la pile servent au traitement de l'interruption. Chaque interruption produit l'exécution d'une instruction RST donnée, et produit à son tour le branchement à un sous-programme spécifique de traitement.

Le retour au programme initial s'effectue par l'instruction RETURN.

De plus, il existe aussi des instructions d'attente qui arrêtent le déroulement d'un programme jusqu'à ce qu'une interruption se produise (instruction WAI dans le 6800 et HLT dans le 8080).

### Initialisation (Reset)

Cette entrée Reset du microprocesseur est utilisée pour le démarrage ou la réinitialisation.

Cette opération a pour rôle de placer le compteur ordinal à une adresse déterminée par le constructeur et de masquer les interruptions.

Dans certains microprocesseurs, l'initialisation remet automatiquement à zéro les registres internes.

Le signal de RESET, dans le 8080, annule le contenu du compteur ordinal. Ainsi, après le RESET, le programme commence à l'adresse mémoire 0. Il est important de souligner que les bascules d'indicateurs (flags), l'accumulateur, le pointeur et les registres ne sont pas remis à 0.

### Interruptions vectorisées ; fin d'interruption

On dit que les interruptions sont vectorisées, lorsqu'une interruption provoque un branchement directement à l'adresse du programme de traitement de celle-ci. Il s'agit dans ce cas d'un adressage indirect.

Par conséquent, la vectorisation consiste à donner les adresses qui apparaissent sur le bus d'adresse après chaque type d'interruption. Par exemple pour le 6800 nous aurons :

| Description des Interruptions | Vecteurs | |
|---|---|---|
| | MSB | LSB |
| RESET | FFFE | FFFF |
| N M I | FFFC | FFFD |
| S W I | FFFA | FFFB |
| I R Q | FFF8 | FFF9 |

L'organisation interne de la mémoire est représentée **figure 7.**

Lors d'un RESET (front montant), le compteur de programme est chargé avec l'adresse contenue dans le pointeur associé à l'entrée RESET (ici FFFF et FFFE). Ces deux adresses permettent de placer sur le bus d'adresse, les deux octets (donc 16 bits) contenus en FFFF et FFFE. Le microprocesseur exécute alors le programme d'initialisation (RESET) commençant à l'adresse contenue dans le compteur de programme.

Du fait de ce mécanisme de « vecteurs », il est important que les programmes mémorisés à ces adresses soient ineffaçables, c'est-à-dire implantés **en mémoire morte.**

Après le traitement d'une interruption, une instruction dans le sous-programme d'interruption permet le retour au programme principal.

## Priorités des interruptions

Pour illustrer le principe des priorités dans les interruptions, imaginons qu'une interruption soit en cours de traitement quand une autre interruption, d'un ordre plus urgent survient. La première interruption sera alors stoppée pour faire place à la seconde. Celle-ci traitée, on reviendra à la première interruption, puis enfin au programme normal.

La **figure 8** représente le diagramme des temps d'un traitement de deux interruptions A et B.

Supposons deux organes A et B reliés à l'unité centrale. Il est pos-

sible d'assigner une priorité d'interruption de l'un par rapport à l'autre, grâce au bit d'état « interruption ».

En effet, après la sauvegarde du contexte dans une pile, suite à une interruption, le bit d'état I est automatiquement mis à 1. Si pour l'un des deux organes (A par exemple), on prévoit au début du programme d'interruption, une remise à zéro de I, une interruption demandée par B pourra être acceptée (car elle n'est pas masquée).

Inversement, si dans le programme d'interruption de B on ne prévoit pas cette instruction tant que l'interruption de B n'est pas entièrement exécutée, un ordre d'interruption de A ne sera pas pris en compte, car cette demande est masquée.

Par conséquent, B est prioritaire sur A.

## Structures à interruptions multiples

Un système d'interruption à plusieurs niveaux de priorité, peut être réalisé soit par programme, soit par câblage.

Le principe d'une interruption

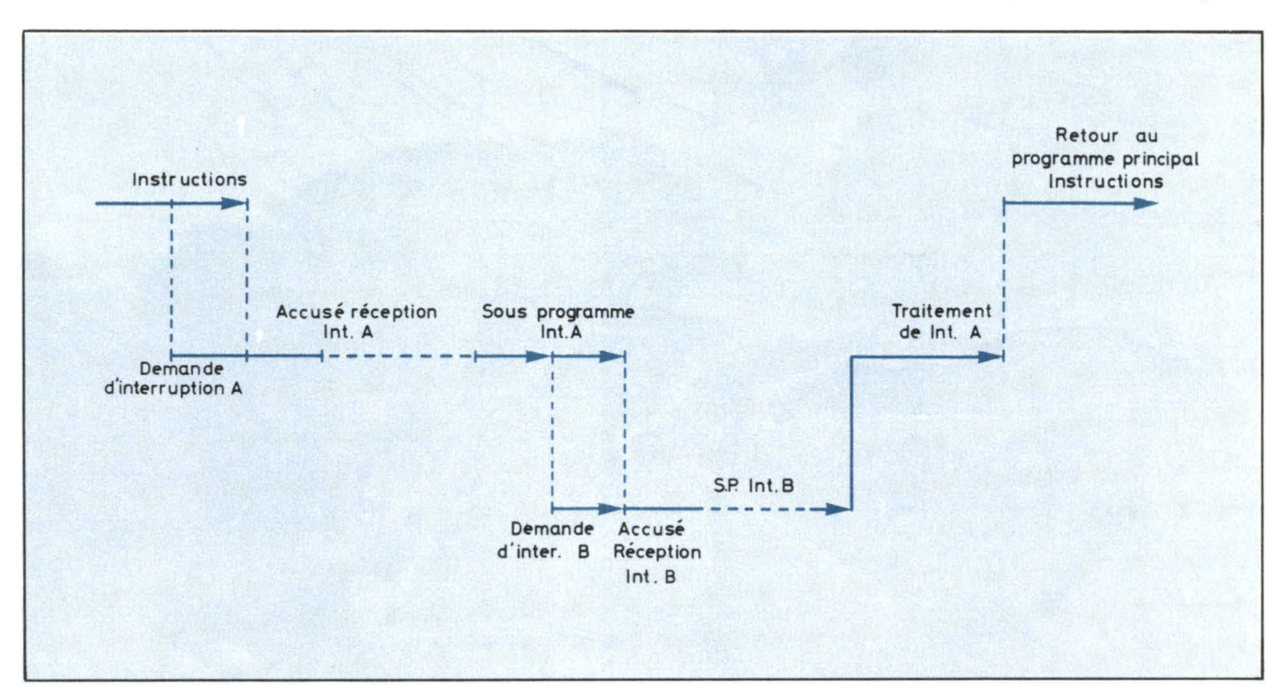
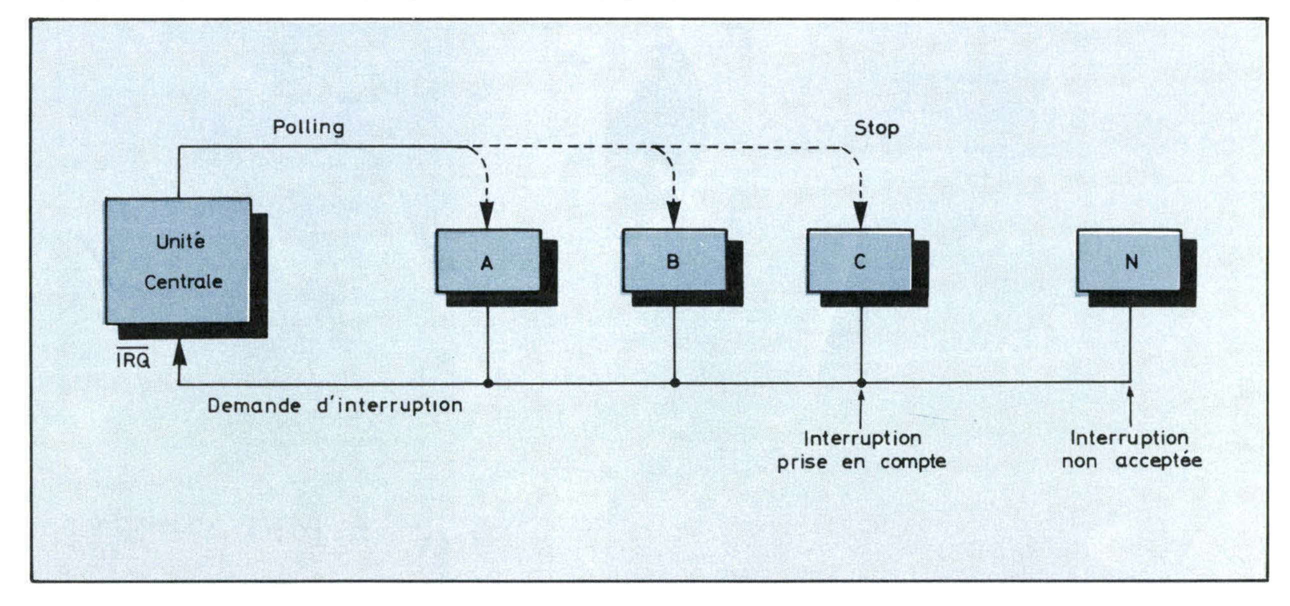



*Fig. 8. – Diagramme des temps d'un traitement de deux interruptions A et B. L'interruption B est prioritaire sur l'interruption A.*

*Fig. 9. – Gestion des interruptions par « polling ». Le programme interroge tour à tour les organes extérieurs et la priorité relative entre chacun d'eux est réglée par leur position « physique » sur le bus d'adresses.*



par programme est le suivant. Dès qu'une demande d'interruption apparaît, l'unité centrale exécute un programme qui interroge tour à tour, les organes extérieurs (Polling), tous les périphériques par exemple.

Dès que la première demande est rencontrée, « l'enquête » s'arrête et cette interruption est exécutée.

On voit que la priorité relative entre les organes extérieurs est simplement réglée par leur « position physique » sur le bus d'adresses **(fig. 9).**

Il est évident que s'il y a souvent des demandes d'interruption, le programme de polling sera de nombreuses fois exécuté, ce qui ralentit le fonctionnement du système.

En ce qui concerne l'interruption par câblage, plusieurs principes peuvent être utilisés. Le plus simple est celui représenté **figure 10.**

Si l'organe B fait une demande d'interruption, la sortie Q de la mémoire est positionnée à 1. Pour toutes les autres bascules $\overline{Q} = 1$. L'information est transmise vers l'unité centrale.

Si par la suite, une demande est formulée par l'organe C, celle-ci ne sera pas transmise car $\overline{Q_B} = 0$. Par contre, si une demande est faite par A, cet organe sera prioritaire.

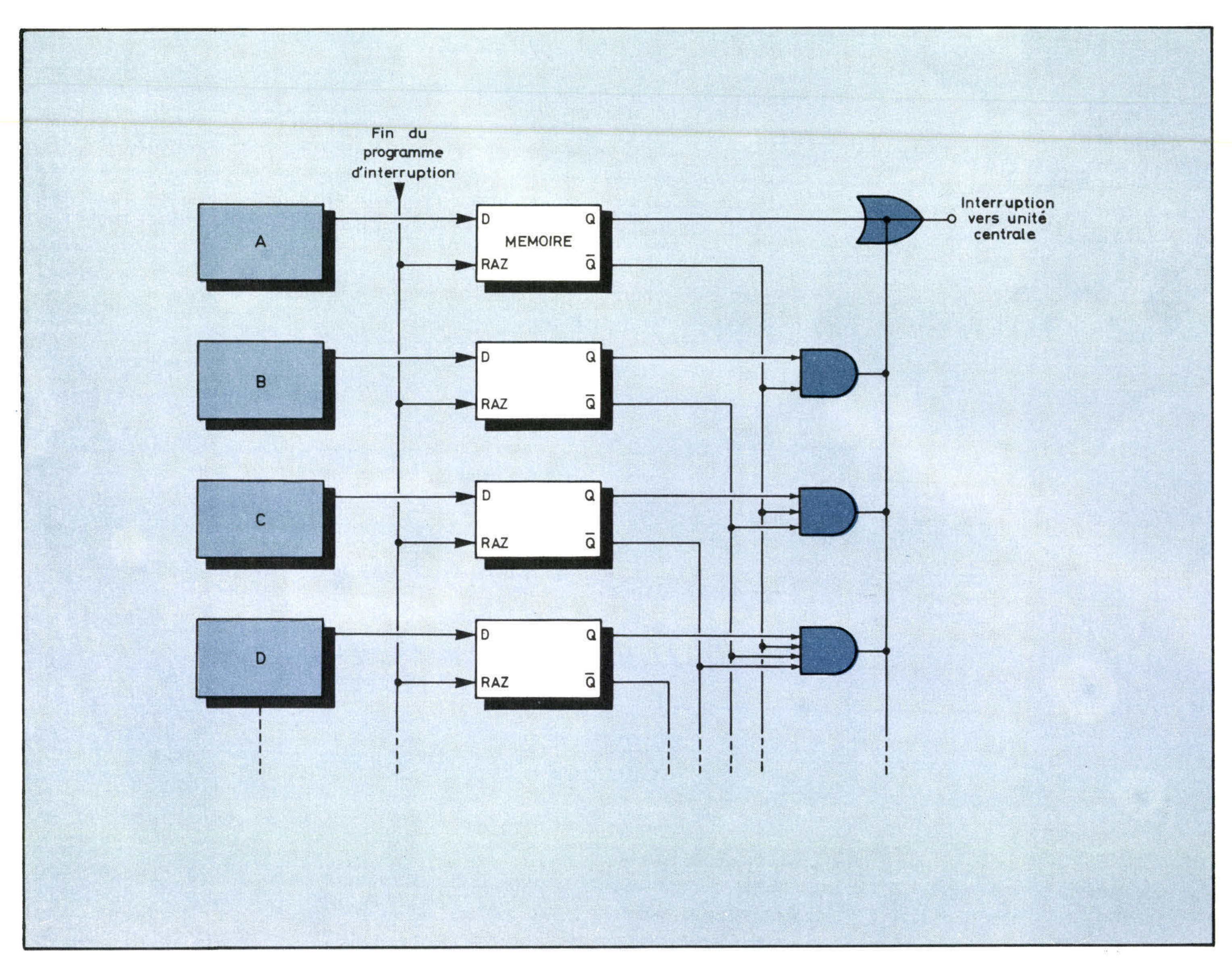
L'exécution du programme de polling sera moins fréquente que dans le cas précédent.

Pour éviter tout câblage compliqué, les constructeurs ont réalisé des circuits intégrés de gestion des interruptions appelé P.I.C. (Programmable Interface Controller). ■

**J. FRÉMAUX**

*Jean FREMAUX appartient à la direction générale de l'Ecole Centrale d'Electronique et assure les cours « microprocesseurs ».*

*Fig. 10. – Principe de la gestion des interruptions par câblage. Il existe des circuits intégrés réalisant cette gestion : les PIC (Programmable Interface Controller).*



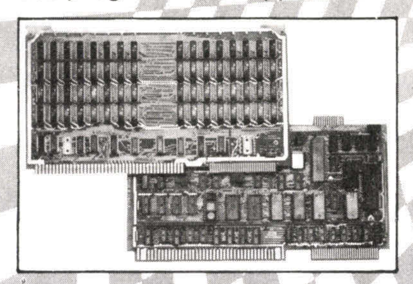
Microordinateur de développement
**PICOLOG 80 D**

**14 550 F*,** version de base
**23 755 F*,** version disque



le confort d'un "Disc Operating System" langage évolué BASIC.
Documentation en français

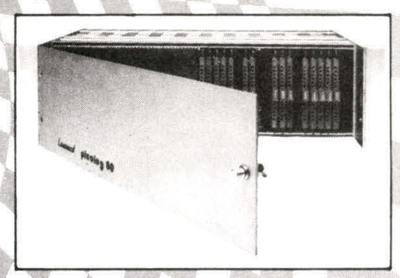
Microordinateur
sur une seule carte
**PICOCARTE 85**
**3 335 F*.** Compatible SBC.
Monotension 5 V. Processeur 8085 capable de recevoir le moniteur de mise au point des programmes d'application.



**LEANORD**

**applications microprocesseurs**

Microordinateur
application **PICOLOG 80**

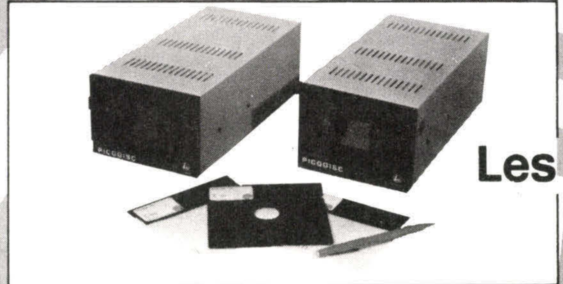


Grande modularité,
plus de 40 coupleurs disponibles.
E/S numériques et analogiques.
Fiabilité de fabrication et contrôle de série, plus de 600 applications réalisées.

Mémoire **RAM,**
carte extension 8K RAM :
**3 360 F*** disponible en :
4K - 16K - 32K - 48K - 64K

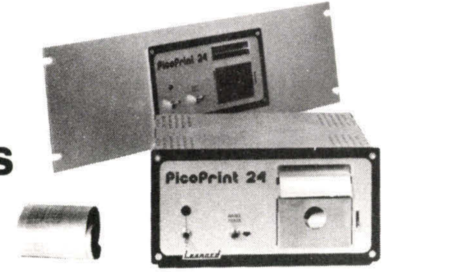
Mémoire **REPROM,**
carte support 16K ou 32K
REPROM : **1650 F***
les plus économiques du marché.

**...le confort...!**





**Les périphériques**
compatibles
tous bus micro




**PICODISC 6 800 F***
Disque souple

**PICOTRACE 9 500 F***
Interface graphique

**PICOPRINT 3 850 F***
Imprimante alphanumérique

EHB birepub 612

* Prix h.t. janvier 79

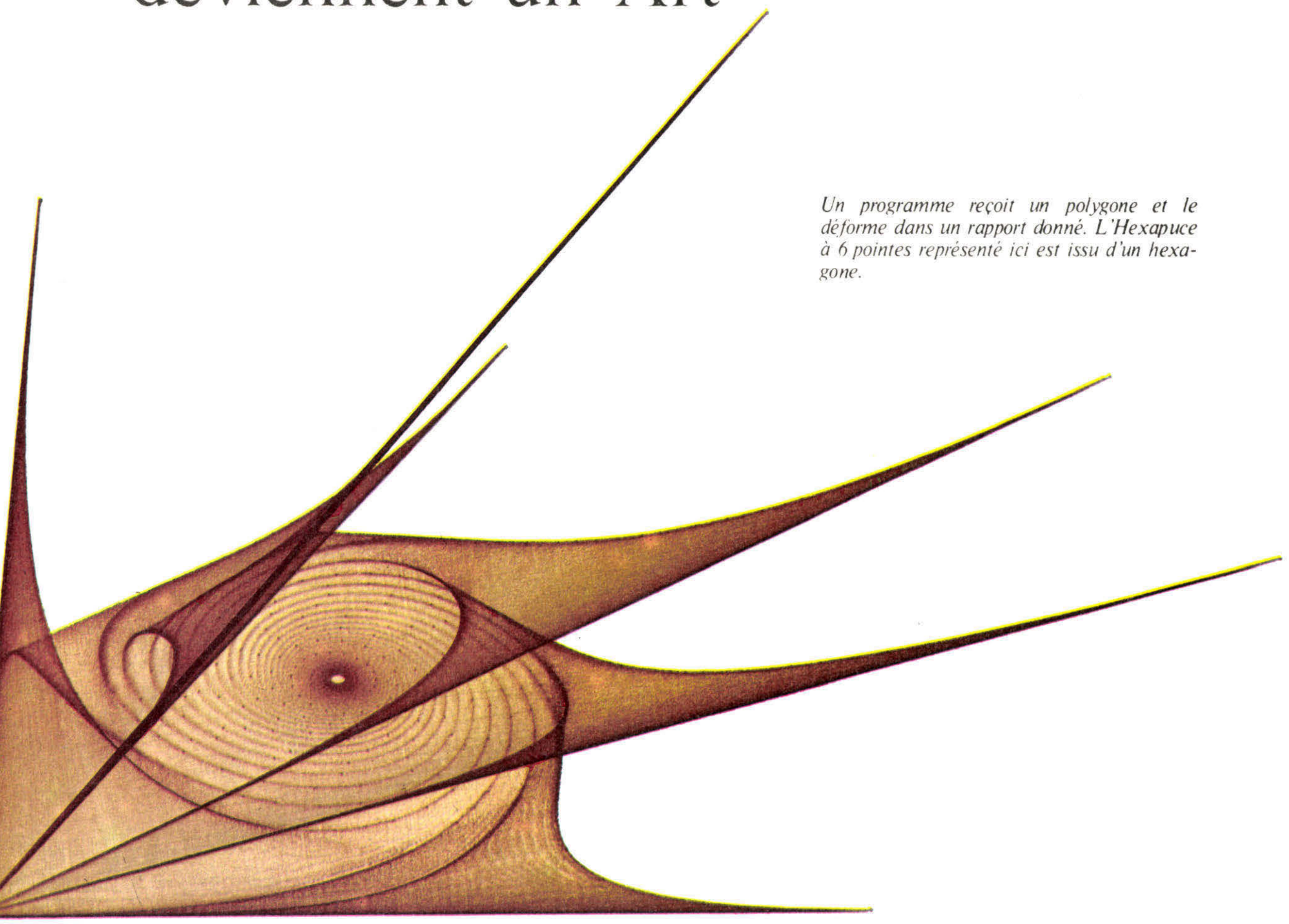


**LEANORD** **PARIS** - 30, route de la Reine - 92100 BOULOGNE - Tél. : (1) 605.63.16 | **LILLE** - 236, rue Sadi-Carnot - 59320 HAUBOURDIN - Tél : (20) 50.43.00 - Télex : 810 910
I.S.A. Groupe Creusot-Loire

**Distributeurs :** Nancy : FACEN (28) 51.00.05 - Strasbourg : FACEN (88) 20.20.80 - Rouen : FACEN (35) 65.36.03

Salon des composants : Allée 9 - Stand 6

# Quand les mathématiques deviennent un Art



*Un programme reçoit un polygone et le déforme dans un rapport donné. L'Hexapuce à 6 pointes représenté ici est issu d'un hexagone.*

La couverture de ce numéro représente une vue partielle du dessin ci-dessus, réalisé à l'I.R.E.M. de Dijon. Il s'agit d'un Institut de Recherche sur l'Enseignement des Mathématiques. Au nombre de 25 en France, ces instituts assurent une formation continue des enseignants, basée sur une liaison constante avec la recherche.

Parmi d'autres résultats de recherche, il y a ces dessins. A proprement parler il s'agit d'une méthode moderne d'enseignement des mathématiques : les étudiants ou élèves sont de plus en plus amenés à travailler avec des consoles graphiques d'ordinateurs dans le cadre de leurs travaux pratiques.

En effet, quelle manière agréable d'aborder le calcul des résidus, les transformées en tout genre (Laplace, Fourrier, etc.), les fonctions dites « spéciales » et les polynômes les plus sophistiqués.

Une conséquence directe de la recherche et du travail sur ces consoles sont ces illustrations graphiques, véritables œuvres d'art par ordinateur.

L'ADAO, association regroupant les gens de tendance artistique comme ceux de l'IREM, innove dans le domaine.

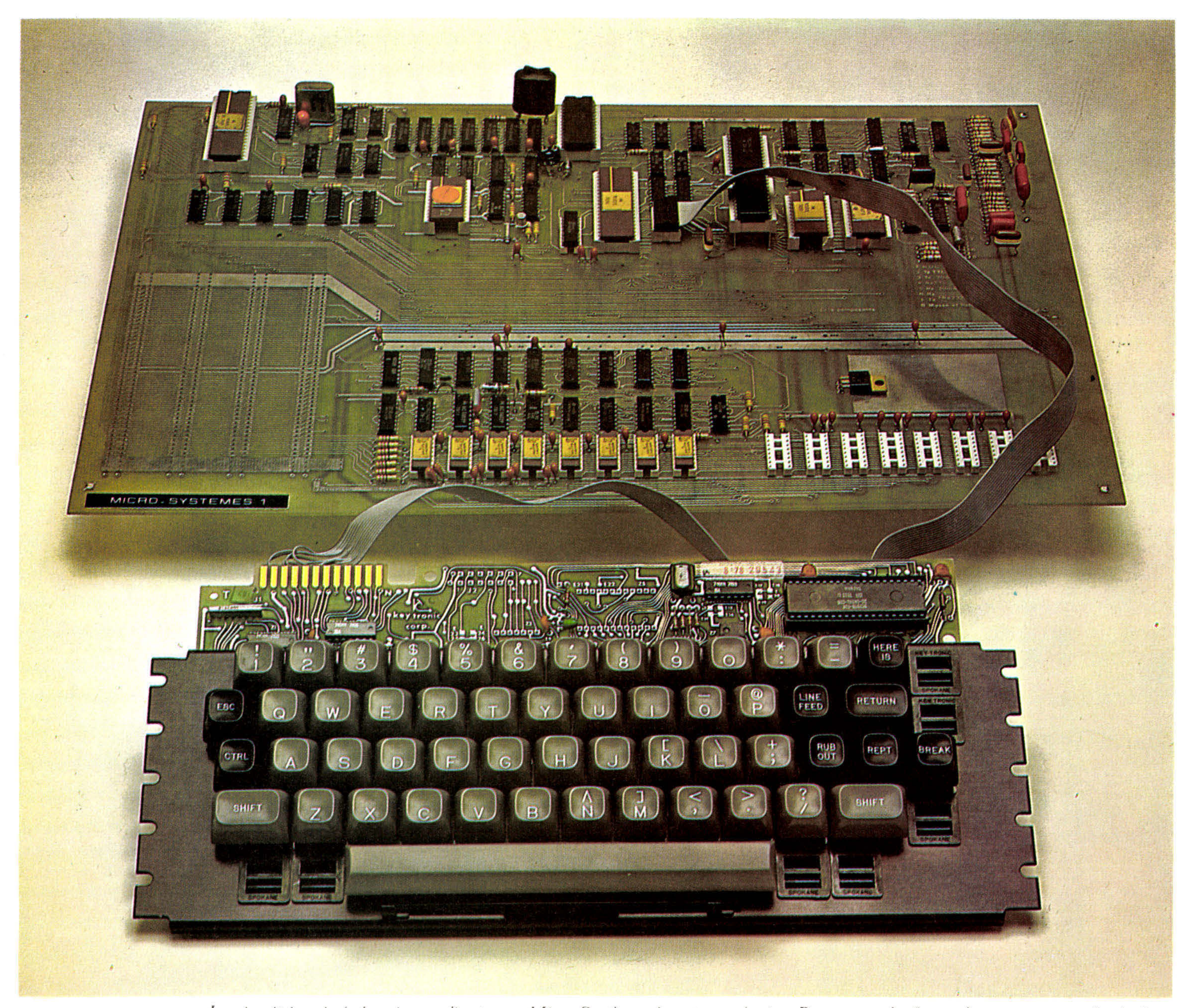
Le dessin analogique (trait continu) nécessite une bonne table X, Y et des fonctions mathématiques complexes. Pour la réalisation de l'**Hexapuce,** le sujet de notre couverture, un programme (laissé en self service dans la boutique d'artisanat du 20e siècle de l'ADAO à Albi) reçoit un polygone et le déforme ensuite, systématiquement, dans un rapport donné. Le polygone de base, pour obtenir ce dessin, n'a pas de surface. Le premier dessin de ce style avait huit pointes, d'où son nom d'**Octopus.** Le suivant en eut sept et s'appela naturellement (OCTO-1) puce, ou Heptapuce (les pointes piquent)... et reçut un prix à la convention informatique 1978. A partir d'un hexagone, voici donc « l'Hexapuce ».

N'oublions pas que ces artistes mathématiciens sont capables de réaliser aussi bien du « Vasarelli » que du « Renoir », si une périodicité ou une mise en forme mathématique évidente le permet. Sont-ils des artistes ou des mécaniciens du dessin ?

Pour en juger, il suffit de regarder de plus près toute leur collection, elle pourrait parfaitement remplir une galerie d'art.

L'A.D.A.O. (Association pour le Développement de l'Art par Ordinateur) est constituée selon la loi de 1901 et met à la disposition des amateurs de micro-informatique un équipement de saisie et de reproduction graphique, un point de rencontre pour tous ceux qui s'intéressent au sujet « Art et Ordinateur », une initiation à l'informatique et au dessin par ordinateur, au Basic ou Logo, langage récursif permettant tout particulièrement le dessin. ■

# Réalisez votre micro-ordinateur « Micro-Systèmes 1 »




*Le circuit imprimé du micro-ordinateur « Micro-Systèmes 1 » et son clavier. Remarquez la façon dont se connecte le clavier sur la carte.*

**Comme nous vous l'avions fait savoir dans notre numéro précédent, l'étude de « Micro-Systèmes 1 » réunit chaque jour davantage d'adeptes, ce dont nous sommes les premiers à nous féliciter.**

**Aujourd'hui, nous ne saurions assez vous recommander d'aborder cette lecture en ouvrant devant vous le dépliant du schéma général de fonctionnement *.**

**Jusqu'à présent, nous avons décrit le fonctionnement des deux blocs les plus à gauche, celui contenant la mémoire vive (RAM) et celui de la logique d'horloge, de l'unité centrale et du contrôleur de DMA.**

**Nous continuerons de la sorte jusqu'à l'épuisement des blocs de ce schéma, chacun faisant l'objet d'un chapitre séparé, malgré une relation étroite de fonctionnement. Ainsi nous étudions maintenant le troisième bloc (B.3).**

**Néanmoins, simultanément à l'étude théorique, vous pourrez entreprendre le câblage et la réalisation proprement dite du micro-ordinateur puisque nous traitons en parallèle la théorie et la pratique.**

* *Micro-Systèmes n° 3, janvier/février, page 35.*

La remise à zéro exige le maintien, à l'état bas de $\overline{RESET}$ pendant au moins six cycles d'horloge.

Avant de nous plonger de nouveau dans les détails du schéma, nous pouvons, d'abord, remarquer ceci :

● Le micro-ordinateur ne possède pas la moindre RAM statique, le montage devant fonctionner avec 16 ou 32 kilo-octets au moins.

● Le rafraîchissement de cette mémoire est concurrent aux demandes d'accès direct à la mémoire (DMA).

La logique d'horloge reçoit deux demandes : REFREQ * et DMAREQ *. La première sera honorée en priorité.

● La visualisation sur un écran T.V. utilise un processeur spécialisé : le contrôleur CRT SF.F 96364 de Sescosem. Il possède sa propre mémoire d'écran, à laquelle l'Unité Centrale n'a pas accès.

La microprogrammation de ce circuit nous évite de gérer le texte sur l'écran. En contrepartie, nous ne pouvons pas « voir » la mémoire de caractère du programme BASIC-utilisateur et, en conséquence, les modifications et les corrections d'erreurs de texte nécessiteront la re-écriture des lignes.

● Un autre processeur fournit au coupleur d'entrée/sortie le code ASCII-7 bits parallèles correspondant à l'enfoncement d'une touche du clavier.

L'Unité Centrale se contente de prendre en compte l'interruption générée par le « STROBE » de ce clavier et range l'octet-caractère qu'on lui présente, après une courte interprétation.

● La présence simultanée de canaux d'entrée/sortie parallèle (PIA 6820-U2) et série (ACIA 6850-U5) permet à un utilisateur disposant de sa propre console de dialogue (visu et clavier) de se passer de certaines parties du câblage (le bloc de visualisation par exemple...).

Ainsi, un programmeur peut utiliser la sortie PIA vers sa propre « visu », sans câbler les autres circuits intégrés du bloc.

De la même manière, dans un premier temps, il pourra ne pas câbler l'interface TTY.

Remarquons toutefois que l'entrée parallèle du coupleur d'entrée/sortie joue un rôle de contrôle déterminant et constitue en quelque sorte l'entrée de la « console-opérateur » (console par laquelle on gère tous les autres périphériques dans les grands systèmes informatiques).

Après ces quelques précisions, détaillons ensemble le schéma. Cette description et la connaissance de la machine vous permettra ensuite de la perfectionner ou, pourquoi pas, de réaliser un autre système.

## B.2 (suite)

Le circuit d'horloge 6875 assure non seulement la gestion des demandes de DMA et de rafraîchissement, mais aussi la remise à zéro et génère les deux phases $\emptyset_1$, $\emptyset_2$ nécessaires au bon fonctionnement du système. Ces deux signaux presque carrés sont indépendants l'un de l'autre.

Une bonne remise à zéro exige le maintien à l'état bas de $\overline{\text{Reset}}$ pendant au moins 6 cycles d'horloge. C'est le rôle de $D_1$ et d'un condensateur au tantale de 4,7 $\mu$F *. La charge lente de ce dernier lors de la mise sous tension assure la bonne initialisation.

Pour alléger un peu notre exposé nous avons porté la description de l'**Unité Centrale 6800** et **du coupleur d'entrée/sortie PIA-6820** dans des articles séparés et indépendants de ce même numéro de Micro-Systèmes.

## B.3 : la mémoire de programme $M_0$ (ROM-BASIC) et son décodage

Cette mémoire doit contenir non seulement l'interpréteur BASIC, mais également des programmes de gestion des ressources * de la plaque.

Lors de l'initialisation du système, par exemple, il faut positionner en entrée ou en sortie les ports du coupleur PIA. Il faut programmer les interruptions, remettre à zéro certains emplacements mémoire ou registres, gérer des coupleurs-série en programmant leurs registres, etc.

Tout ceci est le propre du moniteur. Il doit être contenu dans la mémoire de programme.

Notre système peut fonctionner avec une seule mémoire, $M_0$ ou bien avec une deuxième (DOS), contenant des programmes de gestion des supports magnétiques rapides.

En tout état de cause l'initialisation du système doit s'effectuer de manière automatique sans que l'opérateur ait à remettre à zéro de lui-même tel ou tel mot-mémoire.

C'est pourquoi le décodage de $M_0$ est spécifique à cette mémoire.

Sa grande taille (8 k-octets), en ROM à masque ou en EPROM, permet un adressage facile, assuré par $U_1$, double décodeur deux lignes A, B vers 4 lignes $Y_0$, $Y_1$, $Y_2$, $Y_3$, du type 74139 et par quelques portes ($U_{10}$ et $U_8$) qui assurent le fonctionnement suivant **(fig. 1).**

La première moitié du 74139 **(fig. 2)** reçoit sur les fils de sélection A, B, les fils-adresses $A_{14}$ et $A_{15}$ respectivement. En même temps, la validation à l'état bas qui accompagne les entrées de sélection A, B, est assurée par $\overline{VMA}$ (complément du signal V.M.A. — Valid Adress Memory, indiquant la présence sur le bus-adresses d'une adresse mémoire valide. Ecrit sans barre, le signal est VRAI à l'état haut. Dans le cas contraire, il est vrai à l'état bas et ceci est valable pour tous les signaux).

Ainsi, l'une des quatre sorties $Y_0$, $Y_1$, $Y_2$, $Y_3$, du décodeur passera au niveau bas et correspondra au code binaire sur 2 bits : B, A. Par exemple, 10 produira $Y_2 = 0$ ; 00 sélectionnera $Y_0$, etc.

En cas de non-validation, les sorties $Y_0$, $Y_1$, $Y_2$, $Y_3$ restent toutes à niveau haut.

Deux de ces sorties nous intéressent tout particulièrement : la sortie $Y_2$ (2 $Y_2$, car il s'agit de la moitié « 2 » du 74139, dans les catalogues TTL) valide à l'état bas la sélection de l'autre moitié du 74139, par la liaison entre les broches 10 et 1 ; la sortie $Y_3$ sert à la sélection proprement dite de la ROM BASIC-$M_0$.

** REFREQ : demande de rafraîchissement (refresh request).*

** DMAREQ : demande d'accès direct à la mémoire (DMA request).*

** Fig. 8 p. 47 Micro-Systèmes nº 3.*

** Ressources : ensemble des coupleurs d'entrée/sortie parallèles et séries, mémoire RAM...*

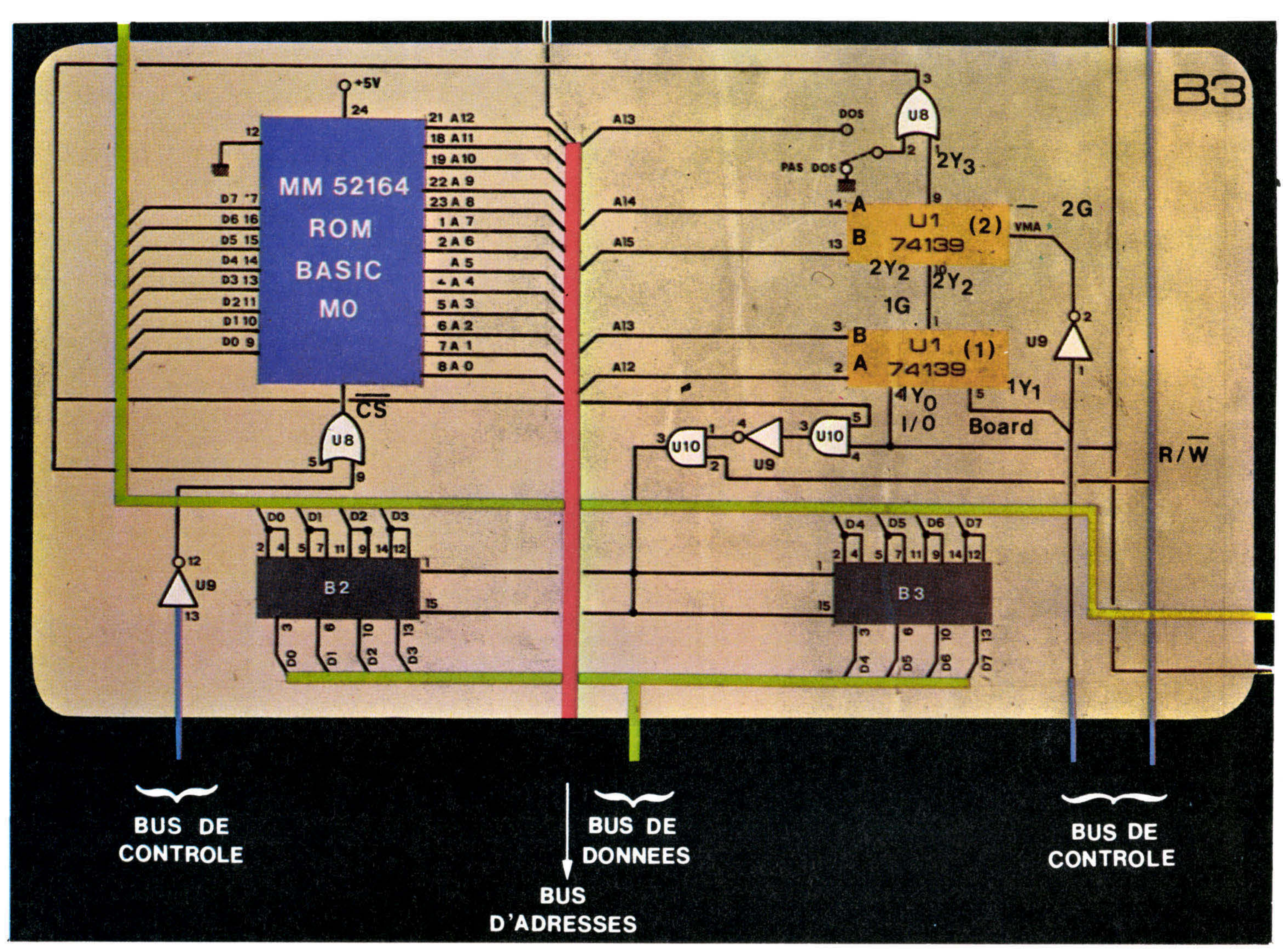



*Fig. 1. – Le bloc 3 du schéma général contient la ROM Basic et son décodage.*

Avec deux fils, $A_{15}$, $A_{14}$, uniquement, on encode toutes les adresses allant de C000 à FFFF, puisque sur un développement en binaire on a :

| $A_{15}$ . . . . . . . . . . . . . . . . $A_0$ |
|---|
| 1 1 X X X X X X X X X X X X X X |

La combinaison 11, correspondant à un « 3 » en binaire, valide $Y_3$. Cette sortie passera à l'état bas quelle que soit la valeur binaire des « X », ce qui signifie que toutes les combinaisons d'adresses allant de :

| | $A_{15}$ . . . . . . . . . . . . . . . . $A_0$ | |
|---|---|---|
| | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | (C000) |
| à | | |
| | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | (FFFF) |

seront concernées par ce décodage.

D'autre part, l'adressage de 8 k-octets demande 13 fils : $A_0$ à $A_{12}$. On constate donc que $A_{13}$ devra départager le bloc de 16 k défini par $A_{15}$, $A_{14}$, en deux blocs de 8 k; allant respectivement de :

| | $A_{15}$ $A_{14}$ $A_{13}$ . . . $A_0$ | |
|---|---|---|
| | 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | C000 |
| à | | |
| | 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 | DFFF |

lorsque $A_{13} = 0$ et E000 à FFFF lorsque $A_{13} = 1$.

Mais, que se passe-t-il si on installe un bloc-mémoire de 8 k dans un espace mémoire de 16 k ?

Deux adresses correspondront au même mot-mémoire, l'une avec $A_{13} = 0$ et l'autre avec $A_{13} = 1$, tant en lecture qu'en écriture.

En particulier :

● Le vecteur d'initialisation (contenu mémoire qui adresse le début du programme d'initialisa-

Les entrées/sorties sont divisées en deux groupes : celles situées sur la carte et celles réservées aux extensions via les connecteurs du bus.

tion) sera situé à deux adresses en même temps :

FFFE, FFFF ($A_{13}$ = 1)
et DFFE, DFFF ($A_{13}$ = 0)

Le microprocesseur force la première combinaison d'adresses. Si $M_0$, qui occupe en mode normal le bloc C000-DFFF, est partiellement encodée, le programme commencera à l'adresse contenue dans les octets DFFE (partie haute de l'adresse) et DFFF (partie basse).

Inversement, une adresse de saut provenant de la mémoire $M_0$ elle-même, produira une exécution dans le premier ou deuxième bloc de 8 k (c'est-à-dire dans $M_0$ elle-même ou dans la mémoire contenant le DOS) en fonction de la prise en compte ou non par $A_{13}$.

En faisant intervenir $A_{13}$, par l'intermédiaire de la porte « OU » $U_8$, on désélecte $M_0$ si $A_{13} = 1$, c'est-à-dire si l'on est dans le dernier bloc de 8 k, E000-FFFF.

Il n'est pas inutile de rappeler que $M_0$ est sélectionné par un seul $\overline{CS}$ *, vrai à l'état bas.

D'autre part, il faut éviter une sélection de $M_0$ en écriture. Ce dernier désagrément est évité grâce à $U_9$ (12-13), qui amplifie et inverse le signal de lecture/écriture (R/$\overline{W}$) du bus de contrôle, de sorte que, lors d'une lecture (R à niveau bas), $M_0$ est validé par la porte « OU » référencée $U_8$. En mode écriture, le signal $\overline{W}$ du bus est inversé et traverse à niveau « 1 » la porte OU pour bloquer $M_0$ par l'entrée de sélection $\overline{CS}$.

Tel est le décodage de la ROM-BASIC.

**Décodage des entrées/sorties**

En ce qui concerne les entrées/sorties, elles sont divisées en deux groupes :

- Les « entrées/sorties carte » pour les boîtiers situés sur la carte.
- Les entrées sorties réservées aux extensions via les connecteurs du bus.

Les « E/S-carte » sont validées à partir de la sortie 4 de la deuxième moitié du 74139. Il s'agit de 1 $Y_0$. Autrement dit, si $A_{12}$, $A_{13} = 0$ et si $A_{15}$, $A_{14} = 10$, pour valider cette moitié du décodeur, il y aura E/S-carte = 0 (l'entrée de validation, broche 1, de cette moitié est commandée par la sortie 2 $Y_2$, broche 10, de la première moitié du 74139).

En composant le code entier on obtient :

| $A_{15}$ | $A_{14}$ | $A_{13}$ | $A_{12}$ |
|---|---|---|---|
| 1 | 0 | 0 | 0 |

c'est-à-dire un 8, ceci fixe en 8000-8FFF les adresses des PIA-s et ACIA-s présents physiquement sur la carte imprimée.

De la même manière, les E/S sur supports bénéficient d'un signal bas de validation pour des adresses $A_{15}$ $A_{14}$ = 10 et $A_{13}$ $A_{12}$ = 01, la broche 5 du 74139 correspondant à 1 $Y_1$. Ce fil rejoint le bus de contrôle et aboutit au bus-supports. L'adresse correspondante sera :

| $A_{15}$ | $A_{14}$ | $A_{13}$ | $A_{12}$ |
|---|---|---|---|
| 1 | 0 | 0 | 1 |

c'est-à-dire 9000 à 9FFF.

Remarquons que 1$Y_2$ et 1$Y_3$, broches 6 et 7, respectivement, sont libres, ce qui nous permettra d'encoder des E/S hors carte aux adresses :

A000-AFFF et B000-BFFF.

Autrement dit, l'espace compris entre

9000 et BFFF

reste à la disposition des E/S utilisateurs.

Attention à ne rien encoder entre 8000 et 8FFF qui représente l'emplacement mémoire réservé à $U_2$, $U_4$ et $U_5$.

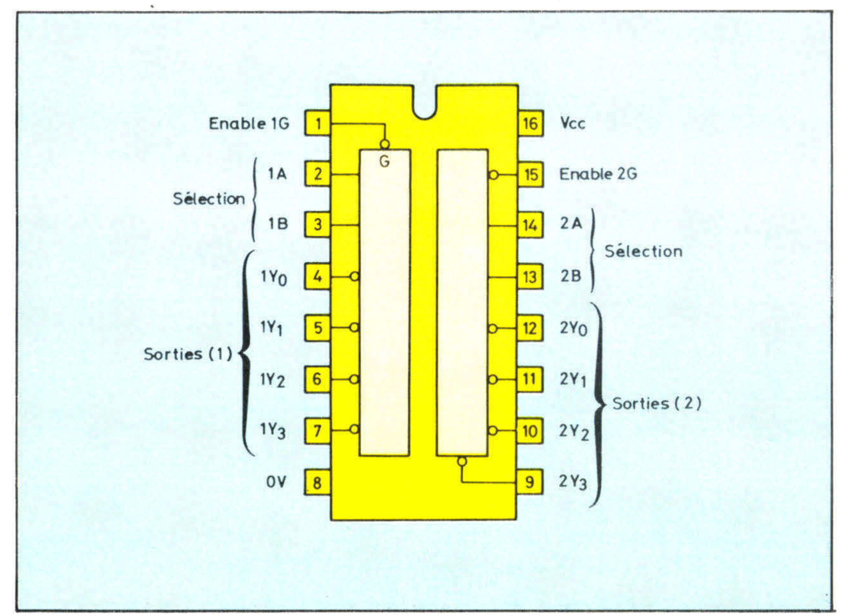
Deux tampons de ligne, $B_2$ et $B_3$, permettent l'amplification du bus de données destinée à $M_0$, à l'interface K7, TTY et à la PIA clavier-visu.

Leur positionnement dans un sens ou dans l'autre est commandé par le R/$\overline{W}$ arrivant sur la borne 2 de $U_{10}$ par la sortie de sélection intermédiaire à la broche 3 de $U_8$ et par la validation $\overline{E/S\text{-carte}}$, issue de la broche 4 du 74139.

Le but des deux portes $U_{10}$ et de l'inverseur $U_9$ employés à cet endroit du schéma est de positionner les tampons vers le microprocesseur si l'on adresse la ROM-BASIC, ou dans un sens commandé par R/$\overline{W}$ si l'on adresse les E/S-carte.

Vérifions le fonctionnement :
Si l'on adresse la ROM-$M_0$ ou le DOS, l'entrée 5 de $U_{10}$ est à zéro, la sortie 3 de la première porte est donc forcément à zéro et la broche 1 du « ET » suivant est à « 1 ». L'entrée 2 décide donc du sens de fonctionnement et c'est précisément le signal R/$\overline{W}$.

*Fig. 2. – Brochage du circuit 74139 : 2 décodeurs-démultiplexeurs, 2 entrées, 4 sorties.*




* *CS : chip select (sélectionne le boîtier).*

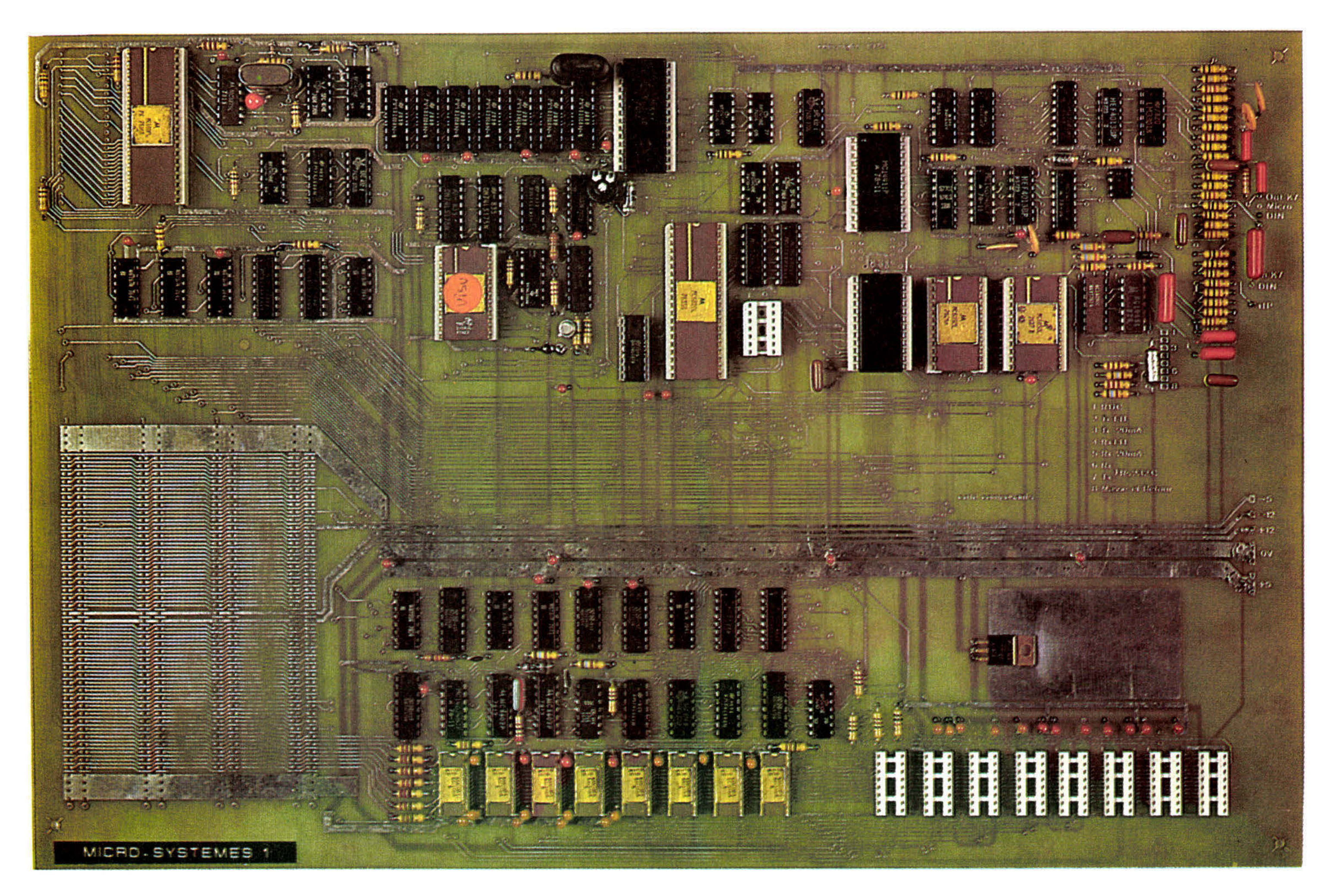



*Fig. 3. – La plaque du micro-ordinateur. Notez la présence des quatre emplacements disponibles pour des connecteurs nécessaires aux extensions éventuelles. Dans cette configuration, le système dispose de 16 k de RAM dynamique, vous pouvez distinguer en bas à droite les huit supports pouvant accueillir 16 k de RAM supplémentaire.*

S'il s'agit d'une lecture, R, $B_2$ et $B_3$ émettent vers le microprocesseur. S'il s'agit de $\overline{W}$, $B_2$ et $B_3$ se positionnent dans le sens inverse.

Tel est donc l'adressage et le décodage de la ROM-BASIC et des entrées/sorties. Nous pouvons le résumer sur le tableau suivant :

| $A_{15}$ | $A_{14}$ | $A_{13}$ | $A_{12}$ | Adresses-hexa | Fonction |
|---|---|---|---|---|---|
| 1 | 1 | 0 | X | C000 - DFFF | ROM BASIC |
| 1 | 1 | 1 | X | E000 - FFFF | DOS |
| 1 | 0 | 0 | 0 | 8000 - 8FFF | E/S - carte |
| 1 | 0 | 0 | 1 | 9000 - 9FFF | E/S - supports |
| 1 | 0 | 1 | 0 | A000 - AFFF | Disponibles |
| 1 | 0 | 1 | 1 | B000 - BFFF | |

**Contenu de $M_0$**

Les vecteurs d'initialisation et d'interruption de $M_0$ pointent aux adresses suivantes :

RESET ........ DE8B
NMI .......... CABA
SWI ........... CAA6
IRQ ........... DE80

En fonctionnement le processeur ira les chercher aux adresses FFFE, FFFF pour le « RESET », FFFC, FFFD pour le « NMI », etc. Si le commutateur DOS-pas DOS ne prend pas en compte $A_{13}$ (pas DOS), l'adressage des vecteurs fera lire le contenu des mots du tableau précédent.

Détailler le fonctionnement des 8 k-octets de programme de la ROM dépasse largement le cadre de cet exposé. Le listing, en « objet », occuperait à lui seul, toute la revue. En assembleur, accompagné de commentaires, il demande un véritable livre d'explications.

Nous reviendrons sur le logiciel contenu dans $M_0$ à propos d'exemples d'applications. ■

# Le montage de « Micro-Systèmes 1 »

Avant même la fin de l'étude théorique du fonctionnement de la plaque du micro-ordinateur, pour que vous puissiez, dès à présent, entreprendre son montage nous pensons qu'il est utile de vous donner des informations indispensables au câblage.

Etant donné le grand nombre de circuits intégrés (88), de résistances et de condensateurs, il est nécessaire d'apporter le maximum de soins lors de la réalisation du système, ce qui simplifiera et réduira de façon certaine le temps de mise au point.

Prenez votre temps, tant pour souder que pour contrôler l'emplacement correct de chaque composant. **Placez des supports de circuits intégrés** chaque fois que cela est possible (en tout état de cause, il est indispensable de mettre des supports pour tous les circuits MOS-LSI : microprocesseur, mémoires...). Mais, attention à la qualité de ces supports, un bon support à réceptacles étamés-dorés coûte souvent aussi cher que le circuit qu'il reçoit.

Pour les circuits TTL, dont le prix est faible, vous pouvez les souder directement sur la plaque.

## Comment implanter

On implante les résistances, condensateurs, circuits intégrés et les supports, **à partir du côté « composants ».**

Malgré les trous métallisés qui nous laissent la possibilité de placer les dipoles indifféremment sur un côté ou un autre, les circuits intégrés ont un brochage qui ne souffre pas une inversion en miroir.

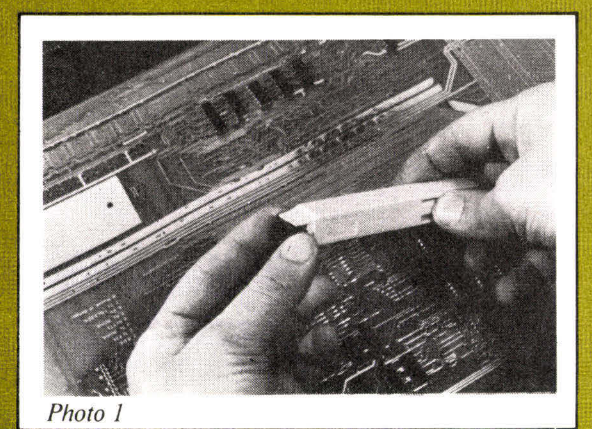


*Photo 1*

Quelle que soit leur taille, 16 broches, 24 ou 40 broches, ils sont munis d'un point ou d'une encoche, à la gauche de laquelle, se situe par convention la broche 1, le circuit intégré étant vu de dessus.

Tous les circuits intégrés seront placés avec l'encoche vers le haut, car telle a été la démarche lors de la fabrication du circuit imprimé (ce qui facilite évidemment le contrôle de l'implantation).

On commence par installer les boîtiers TTL. Ils se reconnaissent à la dénomination 74... ou 8602, etc.

Les boîtiers C.MOS se reconnaissent par un 4000 ou un 14000.

Pourquoi câble-t-on les circuits TTL avant les autres ?

Les boîtiers C.MOS craignent les décharges électrostatiques dues à la moquette ou aux divers objets. Si les boîtiers TTL sont déjà sur place, ils ne laisseront pas en l'air une grande majorité de fils, plaçant des résistances passives entre eux.

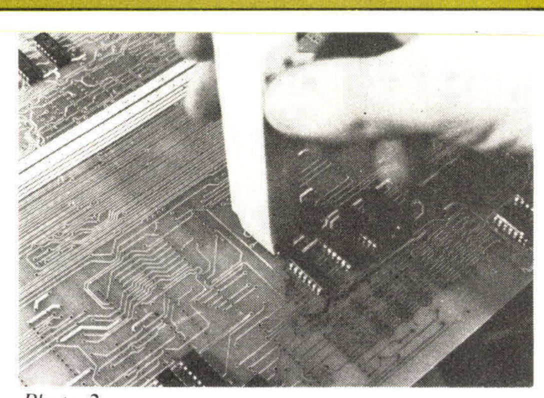


*Photo 2*

Ainsi, les chances de détruire des boîtiers C.MOS diminuent sensiblement.

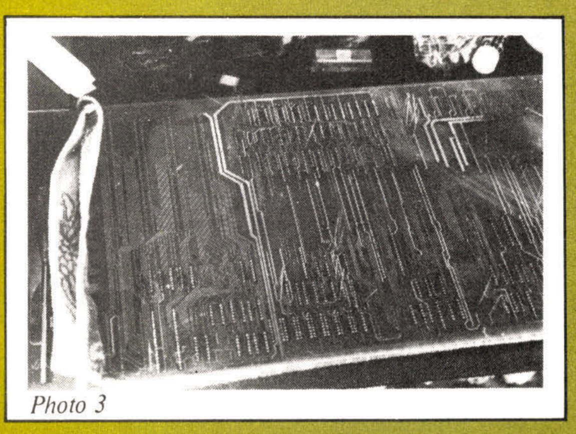
Un outil recommandé pour l'installation sans problèmes de déformation de pattes des circuits intégrés est celui que vous montrent les **photos 1 et 2.** Le nôtre provient de « O.K.-Tool », marque renommée dans le wrapping et l'assemblage de composants électroniques. Divers autres constructeurs proposent des modèles équivalents.

Seul celui qui doit implanter 88 circuits intégrés, parmi lesquels des boîtiers C.MOS fragiles, saura apprécier un tel outil.

Il va sans dire que l'on continuera par l'installation des résistances et condensateurs, en nous servant du schéma d'implantation.

Pour souder tous ces composants un par un, il va falloir, d'une part, retourner la plaque sans ne rien laisser tomber et d'autre part, appliquer à l'aide de mousse ou quelque ressort doux une légère pression sur chaque composant **(photo 3).**

Bien entendu, il est aussi possible de souder un à un les composants.


Photo 3

## Comment souder ?

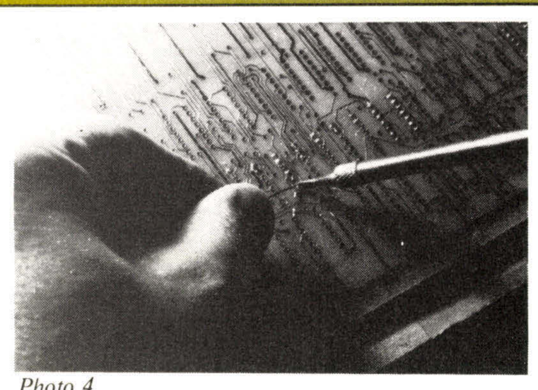
Le fer à souder doit avoir une puissance maximale de 30 W et une pointe très fine. Le diamètre de la soudure ne doit pas dépasser celui des pastilles et trajets du circuit imprimé **(photo 4).**

Dans notre cas, elle ne devra pas dépasser 0,8 mm de diamètre, voir 1 mm au plus.

Ce fil contient dans une âme centrale un flux décapant qui permet d'évaporer la mince couche d'oxyde qui se forme en quelques heures sur toute connexion, même étamée d'un composant électronique.

La soudure devra être brillante et couvrir suffisamment la broche soudée, sans déborder sur les pastilles ou trajets voisins.

Sur la **photo 5** nous montrons quelques soudures, parmi lesquelles une mauvaise : la troisième


Photo 4

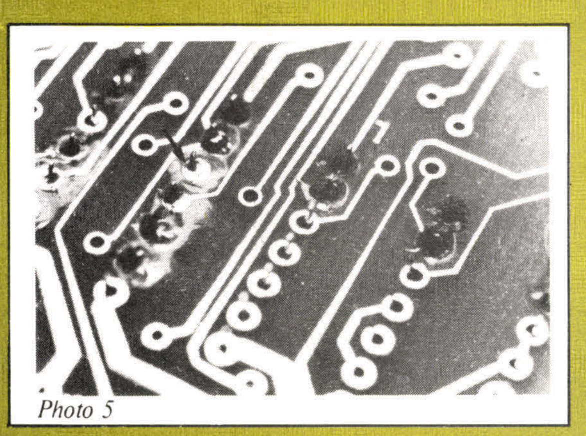
du haut sur la deuxième rangée à partir de la gauche.

Cette soudure deviendra poreuse et bientôt le faux contact qui en résultera ne maintiendra plus mécaniquement la broche du composant, sans parler des parasites que cela pourra engendrer.

## Comment dessouder ?

Si vous soudez directement les circuits TTL sur la plaque, il peut arriver, par malchance, d'inverser un boîtier. Inutile d'essayer de dessouder le circuit intégré à l'aide de tresse ou de pompe à dessouder, le circuit imprimé serait inutilisable et le boîtier irrécupérable.

A moins de posséder une panne de fer à souder en 16 points chauffants au pas DIL (mais comment s'y prendre pour les brochages à 24, voir 40 bornes ?), la solution radicale qui s'impose est de couper très haut, tout près du corps en plastique ou céramique, les broches du circuit intégré, d'extraire les broches une à une avec une pincette et de vider à la pompe à dessouder le contenu du trou métallisé.


Photo 5

## Le contrôle

Avant de mettre votre plaque sous tension, il serait souhaitable de la comparer avec un circuit qui fonctionne. Le test visuel sera beaucoup plus efficace qu'une heure de vérifications.

Des clubs de micro-informatique comme l'AFMI *, Microtel *..., réalisent actuellement avec leurs adhérents **Micro-Systèmes 1** et font des achats groupés de composants. Il serait intéressant, surtout pour les débutants, de prendre contact avec ces associations et bénéficier ainsi de leur aide.

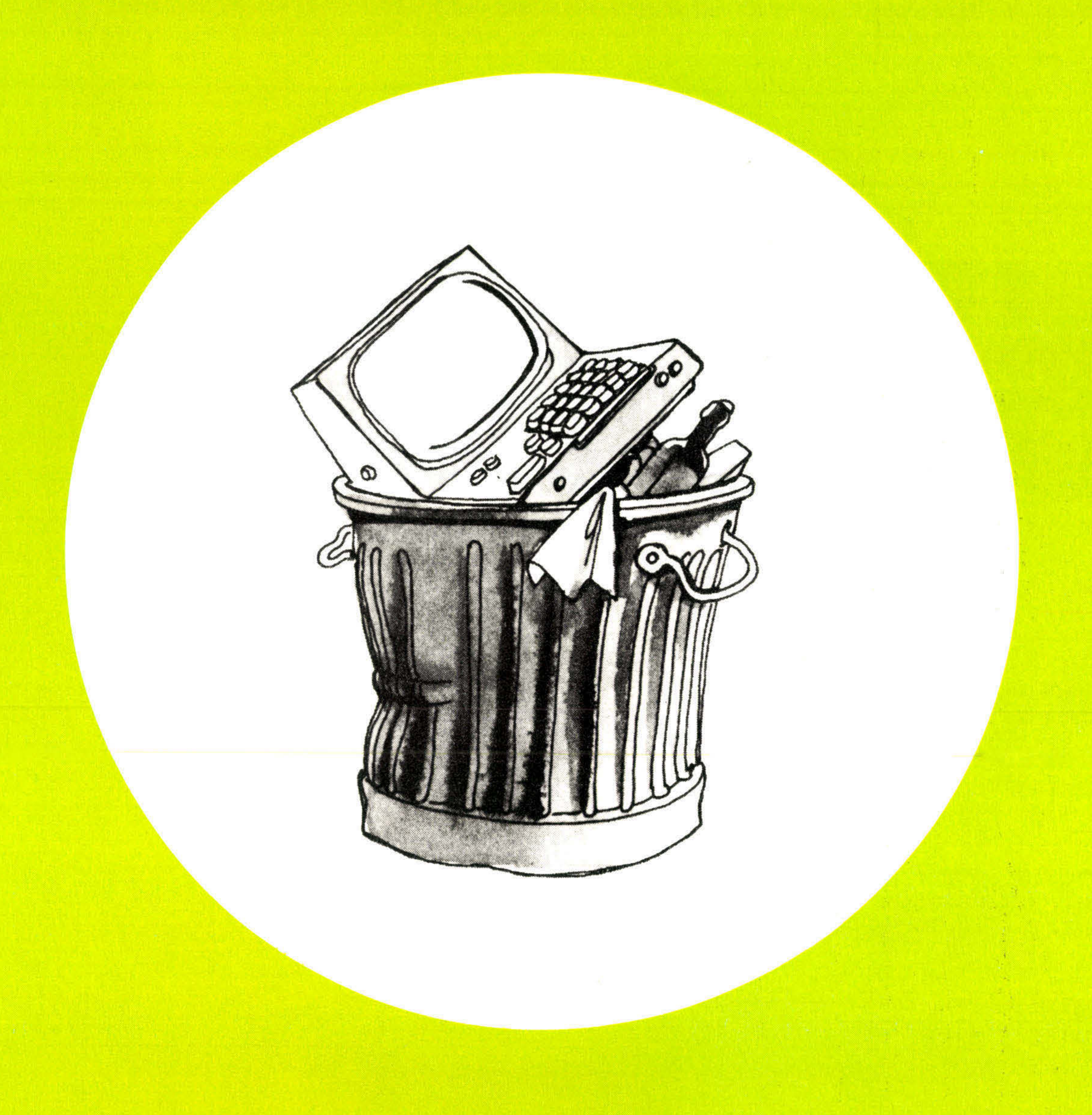
Pour tous les renseignements et conseils concernant ce micro-ordinateur, **Micro-Systèmes** met à votre disposition, tous les mardi et jeudi matin de 9 h 30 à 13 heures, un ingénieur que vous pourrez appeler au 296.46.97. Bien entendu, vous pouvez aussi nous écrire.

Nous faisons notre possible pour que vous soyez nombreux à posséder votre micro-ordinateur chez vous. ■

**A. DORIS**

* *AFMI : tél. 844.77.81.*

* *MICROTEL : tél. 566.37.38.*



# Apprenez-lui à être utile...

Que sait vraiment faire un micro-ordinateur ?
Devez-vous en envisager l'achat ? Lequel ?

# Le Forum de l'Informatique personnelle

Lundi 14 mai 1979, de 14 h à 19 h, Maison de la Chimie,
28, rue Saint-Dominique, 75007 Paris

vous apportera les réponses que vous attendiez.

Cette journée s'adresse à tous les utilisateurs potentiels d'un micro-ordinateur : responsables de PME, ingénieurs, techniciens, étudiants, enseignants, professions libérales.

Vous y apprendrez : à évaluer vos besoins, à connaître ce qui existe, à définir vos options.

Vous saurez quels sont les matériels qui conviennent, ce qu'il en coûte et ce qui se prépare.

## Le programme

**I - Le microprocesseur : cœur du micro-ordinateur**
Architecture de base d'un système. Micro-ordinateurs ou mini-ordinateurs. Définition d'un micro-ordinateur. Avantages des micro-ordinateurs.

**II - Applications personnelles, professionnelles et commerciales**
Calculs scientifiques. Programmes éducatifs. Jeux. Finances. Gestion. Fichiers. Traitement de textes.

**III - Les périphériques**
Claviers. Terminal écran-vidéo. Imprimante. Disque. Bande magnétique. Mémoires de masse futures.

**IV - Choix d'un micro-ordinateur**
Historique des micro-ordinateurs. Critères de sélection. Performance, facilité d'emploi, prix, langage.

**V - Coût d'une configuration**
Coûts réels. Coûts apparents : le logiciel.

**VI - Le choc ordinateur**

**VII - Assistance**
Information et formation à l'usage des micro-ordinateurs.

**VIII - Perspectives d'avenir**

## Le conférencier

Ingénieur E.C.P., docteur ès Sciences (Ph. D) de l'université de Berkeley, Rodnay Zaks est P.-D.G. de Sybex aux U.S.A. et en France, il a personnellement formé plus de 5 000 personnes aux micro-ordinateurs et a présenté récemment des conférences similaires aux Etats-Unis. D'une compétence appréciée aussi bien en France qu'à l'étranger, l'efficacité de son enseignement et l'ampleur de son savoir puisé aux sources mêmes de la micro-informatique américaine vous apporterons les réponses à vos problèmes. Il répondra personnellement à vos questions à la fin du séminaire.

Rodnay Zaks est aussi l'auteur de plusieurs best-sellers sur les microprocesseurs traduits en 10 langues et disponibles en Français.

Organisé par




16/18, rue Planchat
75020 Paris
Tél. : 370.32.75

et patronné par




15, rue de la Paix,
75002 Paris
Tél. : 296.46.97

---

### COUPON D'INSCRIPTION

Inscrivez-moi au Forum de l' « INFORMATIQUE PERSONNELLE » Ci-joint 150 F T.T.C. à l'ordre de SYBEX.

Nom : ________________ Prénom : ________________ Profession : ________________

Société : ________________________________________

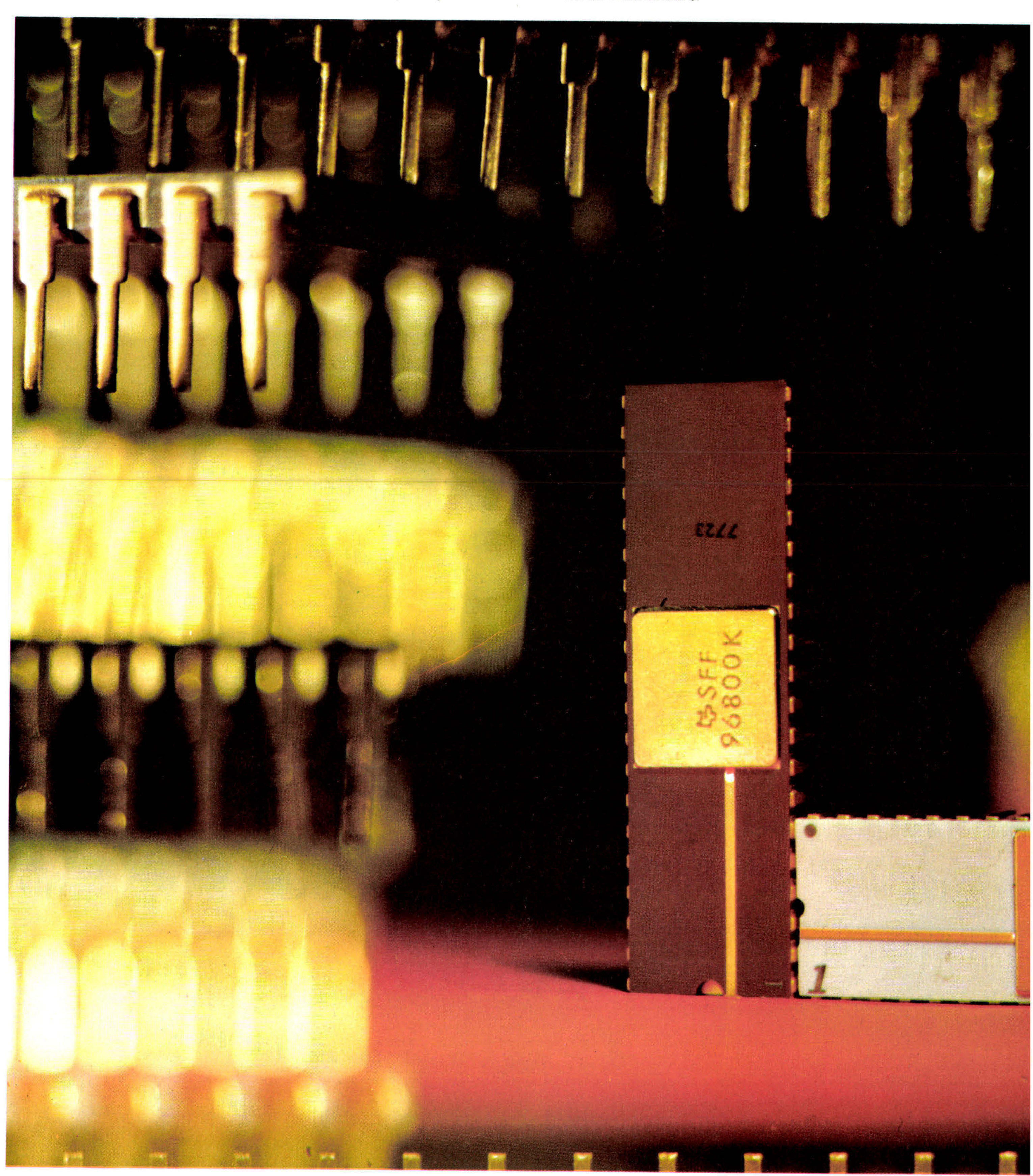
Adresse : ________________________________ Tél. : ________________

Retournez ce coupon à : SYBEX/J.E., 16-18, rue Planchat, 75020 Paris.

MICRO-SYSTEMES N° 4

# Choix d'un microprocesseur

*Photo 1. – Microprocesseurs MOS 8 bits S.F.F. 96800 et bipolaire 4 bits S.F.C. 92901 (cliché Thomson-CSF).*



**Par définition, un micro-ordinateur est un système conçu autour d'un circuit intégré de gestion des informations appelé microprocesseur et doté de mémoires, d'interfaces, de périphériques...**

**Bien entendu, toutes les applications ne requièrent pas nécessairement un micro-ordinateur. Les réalisations simples peuvent souvent se résoudre à l'aide de systèmes minimaux réunissant souvent sur un même circuit imprimé un microprocesseur et une mémoire, ou même être plus facilement réalisables et à plus faible coût grâce à l'électronique câblée.**

**Par contre, des problèmes complexes peuvent amener l'utilisateur à envisager l'acquisition d'un système plus performant comme le mini ordinateur et l'ordinateur.**

**Ainsi, tout au long de cet article, le professeur David tentera de répondre aux deux questions qui se posent à tous ceux qui sont confrontés à ce type de problème : faut-il, actuellement, réaliser le système en logique câblée ou en logique programmée et, si le choix se porte sur la solution du type programmé, utilisera-t-on un mini ordinateur ou un système à microprocesseur ?**

**Dans le cas où l'on décide d'employer un microprocesseur, dans quelle catégorie doit-on le choisir (4 bits, 8 bits, 16 bits, processeurs par tranches...) et dans une catégorie donnée sur quel microprocesseur doit-on porter notre choix ?**

**Certains critères conditionnent ce choix, nous les analyserons en introduisant des notions couramment employées dans l'industrie telles que « l'indice de performance » et les programmes de tests de performances aussi appelés « benchmark ».**

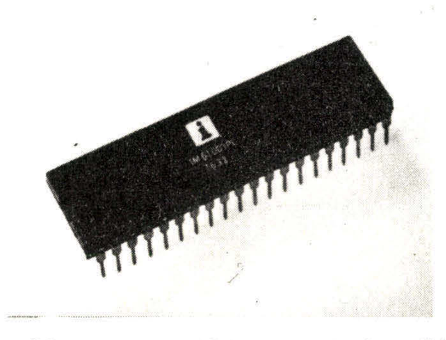


*Photo 2. – Microprocesseur 12 bits en technologie C.MOS : le 6100 de Intersil.*

Les microprocesseurs actuellement sur le marché se distinguent par le nombre de bits de l'unité d'information traitée et par leur technologie de construction.



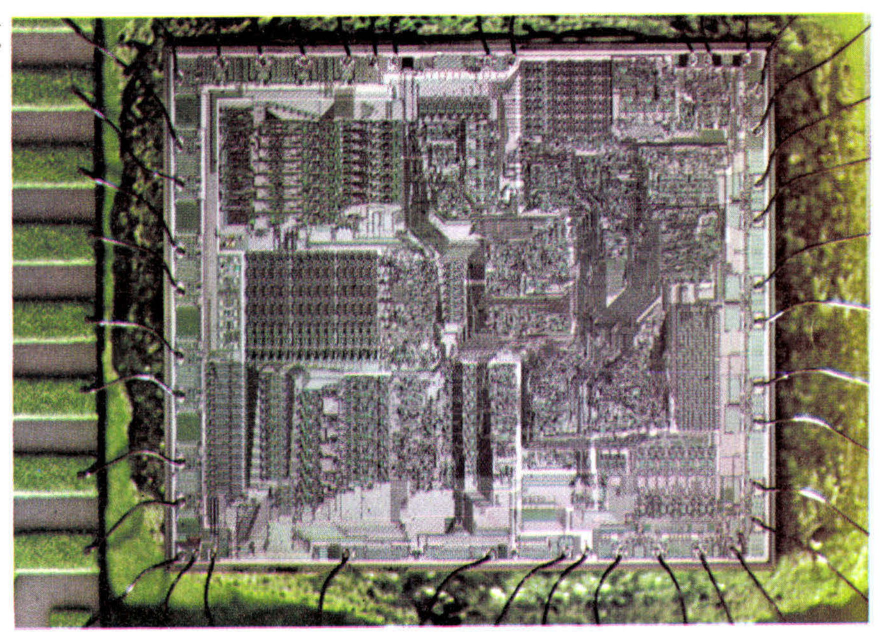


*Photo 3a. – Microphotographie du 2650 de Signetics (Doc. R.T.C.).*

## Microprocesseur contre mini ordinateur

Lorsque l'application envisagée est du type mini ordinateur, l'argument que l'on oppose au microprocesseur, est son manque de performances. En fait, les choses méritent d'être regardées d'un peu plus près.

Dans de nombreux cas, notamment en contrôle de processus dans des laboratoires, on trouve des mini ordinateurs utilisés à une faible fraction de leur puissance. Il est alors évident que l'on a intérêt à les remplacer par des micro-ordinateurs. Dans d'autres cas, un mini ordinateur gère simultanément plusieurs expériences. On peut avoir avantage à le remplacer par plusieurs micro-ordinateurs consacrés chacun à une expérience : le coût total sera probablement inférieur et la plus grande modularité de cette solution apporte plus de souplesse notamment en cas de panne (avec le mini ordinateur tout est bloqué, alors qu'avec N microprocesseurs, seule une expérience est affectée).

D'autre part, les microprocesseurs ne sont pas aussi « malperformants » qu'on pourrait le croire. Certains microprocesseurs 8 bits existent en versions plus rapides (le Z-80 a une version 4 MHz, le 6800 une version 2 MHz, la famille 6500 peut aller jusqu'à 4 MHz) ce qui commence à être respectable.

De plus, des boîtiers annexes existent qui décuplent la vitesse en arithmétique flottante.

Il apparaît de plus en plus de microprocesseurs 16 bits, précisément pour les applications de mini informatique. Enfin, si l'on veut vraiment des performances, on peut recourir aux microprocesseurs par tranches qui conduisent à des systèmes plus coûteux mais calculent en moyenne 5 à 10 fois plus vite que les systèmes précédents. De plus, les courbes du progrès technologique montrent que les performances des micro-ordinateurs ne font que croître.

Là où les micro-ordinateurs sont, pour le moment, inférieurs aux minis, c'est sur le plan des périphériques et, par suite, dans les applications de gestion. Néanmoins, il y a maintenant sur le marché un certain nombre de micro-ordinateurs capables de s'attaquer à des applications de mini gestion. Leur prix les rend accessibles aux particuliers et a fortiori aux petites entreprises. Un des plus remarquables par son rapport performances/prix est le P.E.T. de Commodore...

Dernier argument, peut-être le plus décisif : de plus en plus de mini-ordinateurs bas de gamme apparaissant sur le marché s'avèrent être, en fait, des micros si l'on ouvre le coffret unité centrale. Cette évolution est inéluctable : elle conduit les constructeurs de minis à « se réfugier » dans le marché des « mégaminis » qui, eux, supplantent les ordinateurs bas de gamme tandis que les constructeurs classiques produisent des machines géantes, ou, pourquoi pas, essaient de prendre pied dans le marché des micros.

## Microprocesseur contre logique câblée

La logique câblée s'impose contre les microprocesseurs dans deux cas :

1 - Si l'application est tellement simple que sa réalisation en logique câblée est plus facile que celle d'un système à microprocesseur (dans lequel se posent de toutes façons les problèmes de mise au point d'un programme).

Mais il faut noter que le seuil de complexité de l'application qui justifie l'emploi d'un microprocesseur est très bas. En effet, la complexité et le coût d'une réalisation en logique câblée croissent vite lorsque la fonction devient sophistiquée.

D'autre part, la solution microprocesseur apporte des avantages décisifs :

- moindre coût de réalisation : moins de composants à assembler ;
- moindre coût de mise au point : l'essentiel de la mise au point est celle du programme. En cas d'erreur, c'est le programme qu'il faut modifier, et non un montage électronique comme en logique câblée ;

2 - Si l'application est assez simple pour être traitée en logique câblée, alors la solution logique programmée peut permettre de compliquer le cahier des charges pour ajouter des fonctions au système. Par exemple, la machine à laver à microprocesseur fait plus de choses que la machine à laver simple.

La firme assez imaginative pour ajouter — grâce à un microprocesseur — des fonctions à un appareil classique peut gagner de fortes sommes ;

- la logique programmée permet de réaliser certaines opérations inaccessibles à la logique câblée :
  — tests de vraisemblance en acquisition de données (on dit que cela revient à mettre de l' « intelligence » dans un système) ;
  — application d'algorithmes de régulation complexes en contrôle de processus ;
- le service après-vente des systèmes à microprocesseur est plus facile que celui des systèmes câblés uniquement parce que le système à moins de composants. Un autre avantage est que l'on peut ajouter dans le programme des routines de diagnostics qui testent les fonctions du système et localisent automatiquement les pannes (en allumant des LEDs par exemple) ;
- enfin l'évolution du produit est presque gratuite avec un microprocesseur : si l'on veut ajouter une fonction au système, il suffit le plus souvent de modifier le programme. Il faut donc le remplacer ou ajouter une ROM alors qu'en logique câblée il y a souvent à reconcevoir complètement plusieurs cartes du système.

Le second cas est celui où les performances demandées au système sont telles que, seule, la logique câblée peut y faire face. C'est souvent le cas dans les applications de traitement du signal. Mais ce que nous avons énoncé précédemment s'applique aussi. Avant de dire qu'un système à microprocesseur n'a pas les performances suffisantes, il faut bien examiner toutes les possibilités : microprocesseurs en version rapide, boîtiers arithmétique flottante annexes, microprocesseurs 16 bits, microprocesseurs par tranches. Les performances des microprocesseurs progressent sans cesse offrant chaque jour à de nouveaux domaines la possibilité de profiter de leurs avantages.

Enfin une solution intelligente peut être d'effectuer un prétraitement de l'information en logique câblée, là où le micro-ordinateur n'est pas assez rapide, pour ensuite lui confier les traitements. Un tel cas s'est présenté lors d'une de nos études personnelles où le problème était de déterminer l'histogramme amplitude-fréquence d'un signal (c'est-à-dire le nombre de fois par unité de temps que le signal atteignait un certain nombre de valeurs échelonnées). Les méthodes classiques de traitement informatique du signal ne pouvaient s'appliquer, même avec un gros ordinateur, car les caractéristiques du signal auraient exigé une trop grande fréquence d'échantillonnage et un trop grand nombre de valeurs à emmagasiner et à traiter. En revanche, en mettant en amont de l'ordinateur une série de compteurs commandés chacun par un comparateur, on avait directement les nombres cherchés. L'ordinateur n'avait plus qu'à acquérir et à gérer les valeurs des compteurs, ce qu'un micro-ordinateur peut faire facilement.

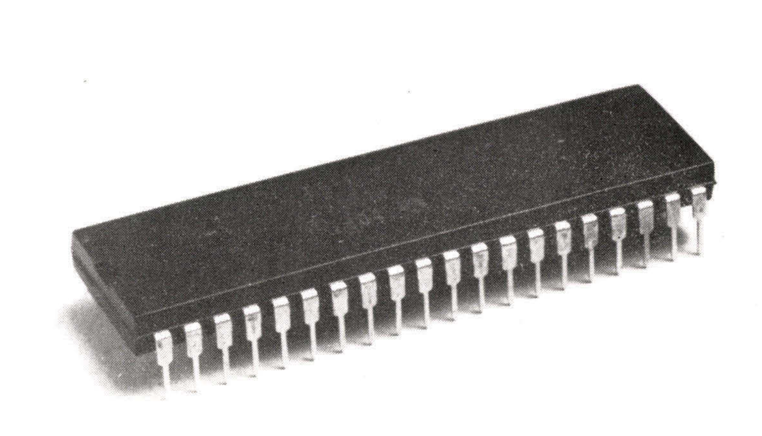


*Photo 3b. – Le 6500, microprocesseur 8 bits, développé par Rockwell, est arrivé en tête des ventes pour l'année 1978 (Statistique Dataquest).*

Un intérêt supplémentaire de la logique programmée dans le contrôle des processus est que, ramenant tous les problèmes de traitement à des algorithmes, elle permet de consacrer plus de temps aux problèmes véritablement spécifiques de l'application que sont la recherche des capteurs qui conviennent aux grandeurs à mesurer dans l'application et la conception des actionneurs capables d'agir sur le processus à contrôler. A la limite, un même modèle de micro-ordinateur standard pourrait servir à des processus extrêmement différents pourvu qu'on lui connecte les capteurs et les actionneurs convenables et que l'on introduise le programme spécifique de l'application.

## Catégories de microprocesseurs

Si le choix d'un microprocesseur donné dans une catégorie est délicat, tant sont voisines les performances des produits de la même catégorie, le choix de la catégorie sur laquelle on doit concentrer son attention est assez simple : une application impose pratiquement celle-ci.

Les catégories de microprocesseurs actuellement sur le marché se distinguent principalement par le nombre de bits de l'unité d'information traitée et par la technologie de construction. Ces deux caractéristiques ont une incidence directe sur les performances qu'on peut espérer du produit et donc sur les types d'applications qui lui seront accessibles.

Il y a cinq catégories principales de microprocesseurs, si on les classe en fonction de la taille des mots.

### Microprocesseurs 4 bits

Ce sont les premiers apparus, en technologie PMOS puis NMOS (Intel 4004, 4040 – Rockwell PPS 4). On peut y rattacher les

micro-contrôleurs (micro-ordinateurs complets) en un seul boîtier (Texas TMS 1000, Rockwell PPS4-1). Ils sont bien adaptés aux traitements décimaux (calculettes, terminaux point de vente) : un chiffre BCD se code en 4 bits, mais ils ont, en fait, été complètement supplantés par les 8 bits qui, par suite de leur baisse de prix, ne reviennent pas plus cher.

### Microprocesseurs monolithiques* 8 bits

Ils forment actuellement le standard du marché (au moins la moitié des applications). Ils conviennent bien aux traitements de caractères (1 caractère tient précisément en 8 bits), au décimal (2 chiffres = 8 bits) à un grand nombre de conversions analogiques numériques (on a rarement besoin de plus de 8 bits ou 0,4 % de précision). Ils existent en technologie PMOS (National Semiconductors SC/MP, Rockwell PPS 8), NMOS (la plus répandue : c'est là qu'on trouve les « tenors » du marché : Intel 8080/8085, Motorola 6800, Fairchild F8, Signetics 2650, Zilog Z-80, MOS Technology série 6500) et CMOS (RCA 1802).

On peut rattacher à cette catégorie les 12 bits qui, pour un coût supérieur, n'ajoutent pas beaucoup d'avantages : il vaut mieux passer au 16 bits. Ils existent en NMOS (Toshiba TLCS 12) et CMOS (Intersil 6100).

Les 8 bits permettent des réalisations qui vont des systèmes minimaux (systèmes complets en 2 boîtiers à partir du F8 ou du MOS Technology 6503, 6530) aux systèmes très élaborés à nombreux boîtiers annexes. Des performances déjà respectables peuvent être obtenues avec les versions rapides des boîtiers NMOS.

### Micro-ordinateurs 8 bits complets en un seul boîtier

Certains systèmes de la catégorie précédente existent maintenant (ou sont annoncés) dans une version en un seul boîtier. Les broches servent non pas à transmettre les bus (inutiles puisque tout est dans le même boîtier) mais elles constituent des broches d'entrées-sorties. On annonce même des entrées-sorties analogiques avec convertisseur incorporé. Ces boîtiers existent en technologie NMOS avec notamment le 6500 de Rockwell, le Mostek 3870 (dérivé du F8) et l'Intel 8048 (dérivé du 8080), dont la version 8748 comporte une EPROM au lieu d'une ROM à masque. Ces systèmes sont employés en masse dans les applications bas de gamme.

### Microprocesseurs 16 bits

Il existe des modèles de cette catégorie depuis quelque temps. Certains d'entre eux ne sont pas en vente sur le marché, ils sont incorporés dans plusieurs mini-ordinateurs. Certains sont commercialisés, comme le Texas Instruments 9900 (64 broches !) ou le PACE de National Semiconductors (PMOS donc lent). Fairchild produit le 9440 (compatible avec le mini : NOVA). Data General en produit aussi une version. Mais les produits qui apparaissent sont probablement plus intéressants : ce sont les versions 16 bits des microprocesseurs 8 bits (ex : Intel 8086). Le « match » 16 bits-8 bits n'est cependant pas forcément gagné d'avance par les 16 bits : à cause de la limitation du nombre de broches, les 16 bits doivent multiplexer certains signaux, ce qui fait

*Photo 4. – Le noyau de la famille d'ordinateur micro NOVA est le microprocesseur mN601 de 40 broches fabriqué par DATA General dans son usine de Sunnyvale en Californie.*

** Par opposition à microprocesseurs par tranche.*

perdre du temps. Il faut donc examiner l'application avec soin. Sinon les 16 bits sont prévus pour les applications haut de gamme et mini gestion.

### Microprocesseurs par tranches

La technologie NMOS permet d'effectuer une addition entre accumulateur et mémoire en 2 $\mu$s (avec une fréquence d'horloge classique). On peut s'attendre au mieux à 0,5 $\mu$s dans les prochaines années. La technologie CMOS est nettement plus lente : il faut compter sur 4 à 5 $\mu$s pour l'opération citée précédemment. En revanche, les avantages de la technologie CMOS sont sa consommation extrêmement faible et son extraordinaire immunité au bruit d'où les applications : équipements qui doivent être capables de fonctionner sur batteries et appareils fonctionnant en milieu industriel très parasité.

Si l'on a besoin d'une rapidité supérieure à la NMOS, il faut faire appel aux technologies bipolaires : la TTL Schottky permet des instructions élémentaires en 200 à 500 ns et l'ECL est la plus rapide (addition en 100 ns), mais elle est délicate à utiliser. L'ennui des technologies bipolaires est qu'elles ne permettent pas une aussi grande densité d'intégration que les MOS. On ne peut donc pas fabriquer en bipolaire des microprocesseurs équivalents à ceux cités précédemment.

Il faut faire appel à une autre architecture dans laquelle on intègre en un boîtier tout le chemin des données concernant une tranche de n bits. Si l'on veut constituer une unité centrale de p x n bits, il suffit de juxtaposer p tranches. Les boîtiers ont en effet des broches d'entrée et sortie de retenues qui permettent une telle juxtaposition (les microprocesseurs classiques ne sont pas juxtaposables car ils n'ont pas ces broches). On voit donc qu'il faut plus de boîtiers qu'avec les microprocesseurs classiques, d'autant qu'il faut aussi ajouter de la logique d'enchaînement des **(micro)** instructions. On obtient donc des systèmes plus complexes et plus coûteux réservés aux applications très rapides (systèmes de transformation de Fourier rapide, unités centrales de certains minis : l'Interdata 6/16 et le Mitra 105 font appel à des 2901).

En TTL LS le marché est dominé par la tranche de 4 bits AMD 2901 : de nombreux constructeurs en sont la seconde source et il y a toute une famille associée à ce circuit. En ECL, Motorola produit la tranche de 4 bits 10800. Enfin Signetics a lancé un « contrôleur de communications » bipolaire qui n'est ni une tranche ni un microprocesseur, le SMS 300.

## Quel microprocesseur ? Critères de choix

Une fois décidée la catégorie de microprocesseur qui convient à l'application, il faut encore déterminer quel microprocesseur sera choisi. Nos remarques s'appliqueront surtout à la catégorie des microprocesseurs monolithiques 8 bits, puisque c'est la plus importante à l'heure actuelle (50 % du marché, ou plus), et que c'est celle où le choix est le plus vaste et le plus délicat.

Les fiches techniques des constructeurs ne sont pas sans rendre ce choix encore plus délicat : elles insistent sur les points forts du produit et « glissent » sur les points faibles. Il faut donc les étudier avec soin et d'un œil critique. Par exemple, la fiche technique insiste sur une fréquence d'horloge admissible élevée sans dire combien il faut de cycles pour effectuer une opération. C'est ainsi qu'un Z-80 avec horloge à 2,5 MHz et un 6502 avec horloge à 1 MHz ont à peu près les mêmes performances. Un 6502 avec horloge à 2 MHz va plus vite qu'un Z-80 avec horloge à 4 MHz.

Enfin, il faut bien admettre que le choix dans la catégorie des 8 bits est très délicat pour la simple raison que les différents produits ont des performances sensiblement voisines ; chacun a des avantages particuliers que les autres n'ont pas ; aucun n'a tous les avantages.

Passons maintenant en revue les principaux critères de choix qu'on peut avancer. Bien entendu, ce qu'on veut optimiser c'est le prix de revient à l'unité du système que l'on veut concevoir soit :

$$PR = \frac{F}{N} + V$$

où N est le nombre d'exemplaires du système que l'on envisage de produire ;
F sont les frais fixes (indépendants de N), dont la composante principale est le coût de développement du produit ;
V les frais variables (valables pour chaque exemplaire), par exemple le coût intrinsèque des boîtiers qui font partie du système.

Les différents critères jouent soit sur V, soit sur F. Ils ont donc une importance qui varie du tout au tout selon que N est petit (système de faible diffusion tel que automatisme de laboratoire) ou grand (produit grand public). Nous examinerons d'abord les critères qui peuvent être considérés comme agissant à la fois sur F (coûts de développement) et sur V (coûts intrinsèques). Ce sont les critères de performance comme par exemple le jeu d'instructions : un jeu d'instructions favorable rend les programmes plus faciles à écrire ce qui diminue les coûts de développement et il permet un meilleur rendement dans l'utilisation de la mémoire, ce qui peut économiser des boîtiers mémoire d'où un moindre coût intrinsèque.

## Critères de performance

Les critères que nous allons donner ne s'appliquent qu'à condition que les caractéristiques générales de la catégorie et de la technologie envisagées nous placent dans la gamme de performances voulue pour l'application. En somme, ces critères jouent « au second ordre ».

### Cycle d'horloge

Ce paramètre conditionne directement la vitesse d'exécution, mais il faut l'examiner avec pré-

La meilleure manière de comparer les performances de plusieurs microprocesseurs est de leur faire exécuter un même programme dit programme de « Benchmark ».

cautions : il ne suffit pas de définir le cycle d'horloge le plus rapide admissible, il faut aussi voir combien de cycles il faut pour effectuer une opération donnée. Ce critère joue le mieux lorsqu'il se présente sous la forme : choix de versions plus ou moins rapides du même processeur ; on peut alors choisir la version qui est la mieux adaptée à l'application. Les coûts de programmation sont diminués lorsque, sachant que le microprocesseur a des performances suffisantes, on n'en est pas à chercher à économiser des cycles par-ci par-là dans le programme.

Une précaution à prendre lorsque l'on cherche de hautes performances et qui joue beaucoup sur le coût intrinsèque est la suivante : si l'on prend un processeur à cycle rapide, il faut que les mémoires et les autres boîtiers associés soient à l'avenant, ce qui augmente fortement le coût. Le véritable paramètre à prendre en considération est donc plutôt :

**IP** = Indice de Performance

$$= \frac{\text{temps nécessaire pour effectuer une opération donnée}}{\text{temps d'accès mémoire maximum exigé}}$$

Par exemple, le Z-80 (2,5 MHz) et le 6502 (1 MHz) ont sensiblement les mêmes performances, c'est-à-dire le même numérateur. Mais le Z-80 exige un temps d'accès mémoire de 450 ns alors que le 6502 se contente de 575 ns soit un IP meilleur de 30 %. Il faut noter qu'ici cela joue peu, les mémoires de 450 ns n'étant pas beaucoup plus chères que celles de 575 ns de temps d'accès. Mais si l'on compare le Z-80 4 MHz qui exige 270 ns de temps d'accès au 6502 2 MHz qui se contente de 300 ns tout en allant plus vite cela peut commencer à compter.

### Jeu d'instructions et modes d'adressage

Ce critère est le plus subjectif de tous : chacun peut avoir ses préférences. Ceci explique qu'il donne lieu aux discussions les plus passionnées. En fait, les principaux microprocesseurs se valent sensiblement dans ce domaine : sur l'ensemble d'un programme, ce qu'un processeur perd par rapport aux autres, il le regagne un peu plus loin. Il faut se méfier des instructions style gadget mises en avant par certains, et que personne n'utilise parce qu'en fait, elles ne sont pas pratiques. Le nombre de registres internes peut faire impression mais il faut en examiner l'utilité réelle. Personnellement, nous accorderions plus d'importance aux modes d'adressage.

Ce qu'on peut dire en définitive c'est que, si le processeur envisagé a un jeu d'instructions du style mini ordinateur, un programmeur qui a fait des études d'informatique classiques se sentira plus à l'aise avec lui alors qu'il jugera rébarbatif le jeu d'instructions des processeurs qui ont plutôt le style « logique câblée adaptée » (ex : F8, RCA 1802).

Le jeu d'instructions influe fortement sur les coûts de développement. Il n'a donc que peu d'importance en production de masse. Les modes d'adressage agissent sur les deux sortes de coûts (développement : de bons modes d'adressage facilitent la programmation ; coûts intrinsèques : de bons modes d'adressage entraînent une utilisation plus efficace de la mémoire ce qui permet des économies).

On peut rattacher à ce critère la façon dont le microprocesseur gère les interruptions : cela peut être important dans certaines applications.

### Programmes de tests de performances (Benchmarks)

La meilleure manière de comparer les performances de plusieurs microprocesseurs est de leur faire exécuter un même programme dit programme de « Benchmark » * écrit de façon à être assez général pour tester toutes les possibilités des machines ou au contraire, de façon à être représentatif de l'application envisagée.

Mais ce problème de la représentativité est source d'interminables discussions. En fait, il faut proscrire **les benchmarks** écrits par un constructeur : en effet, celui-ci a investi un temps considérable à optimiser son programme en utilisant les particularités de son microprocesseur et en évitant soigneusement ses faiblesses. L'utilisateur doit faire appel à un benchmark écrit par lui-même sans y consacrer trop de temps et en s'en tenant à ses habitudes de programmation : il aura ainsi un résultat représentatif de ce qu'il pourra « tirer » de chacun des microprocesseurs envisagés.

## Critères qui influent sur le coût intrinsèque

Les critères de cette catégorie sont plutôt hardware puisqu'ils concernent le prix de revient de la construction du système.

### Prix du microprocesseur

Nous ne citons ce critère que pour l'écarter immédiatement. Le prix du microprocesseur lui-même est négligeable vis à vis du prix total du système. D'autre part, la plupart des microprocesseurs d'une catégorie ont des prix voisins. Par exemple, il y a quelques années, National Semiconductors a lancé un microprocesseur le SC/MP, de performances nettement inférieures aux autres 8 bits, mais censé être moins cher. Le problème est que les autres microprocesseurs de la catégorie sont plus rapides, ont un jeu d'instructions beaucoup plus riche et un coût à peine plus élevé.

Ce qui intervient beaucoup plus dans le prix du système constitué, c'est le prix des boîtiers annexes.

** Benchmark signifie « repère » en anglo-saxon.*

## Prix des boîtiers annexes

Un système complet peut comporter un assez grand nombre de boîtiers annexes (logique d'appoint, boîtiers d'interface). Ceux-ci ont donc tendance à compter plus que le prix du microprocesseur lui-même d'autant que les constructeurs tendent à baisser le prix du microprocesseur et à se « rattraper » sur les boîtiers périphériques. Il ne faut pas se laisser prendre à ce jeu.

On a intérêt à donner la préférence à une « famille » comportant un bon choix de boîtiers d'interface disponibles et bon marché. Une des familles les plus riches à ce point de vue est celle du **6500.** Il est intéressant que parmi ces boîtiers de l'interface figurent des boîtiers multifonctions associant par exemple RAM, ROM, ports d'E/S, temporisateurs programmables. Cela permet de constituer des systèmes minimaux à très faible nombre de boîtiers. La limite dans cette direction est le micro-ordinateur en un seul boîtier très favorable pour les produits de grande consommation. Pour les systèmes moyens, il est très commode de disposer de plusieurs choix possibles, permettant une adaptation optimale à l'application.

Le problème du choix des mémoires permis par le microprocesseur et déjà décrit précédemment est à rappeler ici aussi.

## Facilité d'emploi hardware

On peut grouper ici différents critères tels que :

- nombre de boîtiers nécessaires pour constituer le système,
- complexité des signaux d'horloge utilisés,
- horloge incorporée dans le boîtier,
- consommation (à l'intérieur d'une technologie donnée),
- nombre d'alimentations nécessaires.

Les critères précédents ne prêtent pas à discussion. D'autres sont plus subjectifs comme :

- facilité du D.M.A. (il faut d'ailleurs examiner soigneusement si celui-ci est vraiment nécessaire dans le système que l'on conçoit. Le D.M.A. est en général d'autant moins nécessaire que le microprocesseur est plus performant) ;
- différents signaux de contrôle fournis. C'est là que les goûts s'opposent le plus. Nous sommes personnellement en faveur des signaux les moins nombreux et les plus simples possibles, à condition de permettre toutes les possibilités ;
- possibilités particulières. Par exemple le Z-80 commande lui-même le rafraîchissement des mémoires dynamiques. Malheureusement, une étude soignée montre que tous les problèmes ne sont néanmoins pas résolus.

On peut dire en général que, dans ce domaine, les microprocesseurs qui sont apparus plus tard comme améliorations du 8080 (Z-80 et 8085) ou du 6800 (6500) sont en effet plus favorables que leurs devanciers.

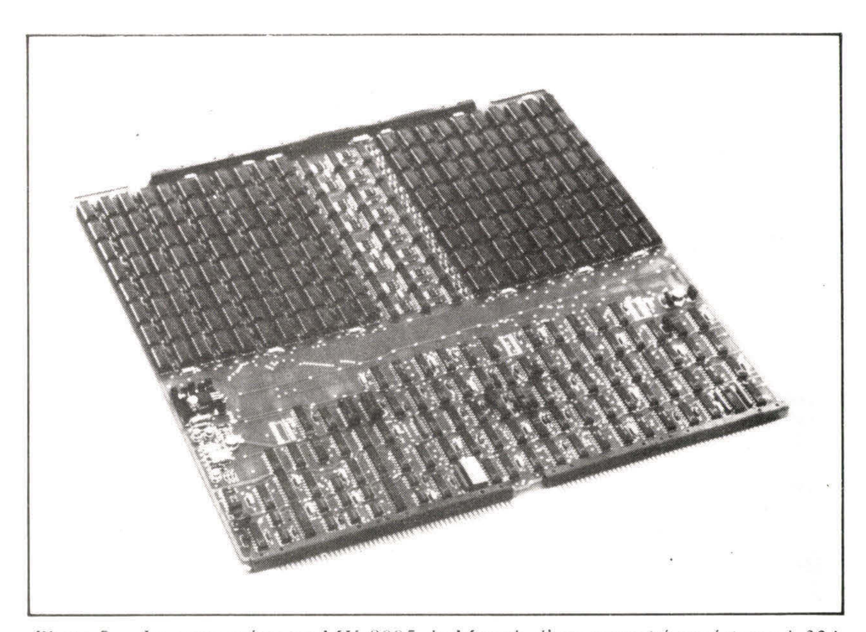


*Photo 5. – La carte mémoire MK 8005 de Mostek d'une capacité supérieure à 32 k x 16 bits.*

## Disponibilité Secondes sources

Il est tout à fait fondamental, en production de masse, de ne pas se heurter à des délais de livraison trop longs. La présence de secondes sources (c'est-à-dire de constructeurs différents qui fabriquent le même produit) est une garantie dans ce domaine, d'autre part, c'est également une cause de baisse de prix car elles se livrent souvent à de véritables « guerres des prix ».

Il faut noter que le comportement du premier constructeur vis-à-vis de ses secondes sources est différent dans le monde des microprocesseurs de ce qu'il est pour les circuits ordinaires. Pour les circuits ordinaires, les secondes sources ne sont que des concurrents. Dans le monde des microprocesseurs, les secondes sources sont des concurrents qui de toutes façons prendront au premier constructeur une partie de son marché, mais, par ailleurs, avoir des secondes sources constitue une consécration pour la conception du produit, et elle lui donne de la crédibilité.

Il y a, dans les microprocesseurs, des secondes sources officiellement autorisées, et d'autres qui sont non autorisées : par exemple les Japonais (NEC) ont fait une copie au microscope électronique du 8080 (Intel n'a pas pour politique d'autoriser les secondes sources).

En résumé, disons qu'il ne faut pas classer les microprocesseurs en fonction du nombre de leurs secondes sources, mais qu'il est plus rassurant de choisir un produit qui en ait.

La facilité d'emploi du support logiciel est un des critères-clé de sélection et permet de gérer au mieux le système de développement.

## Critères qui influent sur les coûts de développement

La principale opération dans le développement d'un produit étant l'écriture et la mise au point du programme associé, les critères qui interviennent ici concernent le software.

Rappelons pour mémoire le jeu d'instructions et les modes d'adressage. Ce critère est fondamental pour faciliter l'écriture du programme. Notons toutefois, qu'il intervient peu pour un produit de grande diffusion où le programme est en général assez simple et où le temps passé à écrire le programme est à diviser par le nombre d'exemplaires. Il est en revanche crucial pour une application à faible diffusion mais à programme très complexe.

Les autres critères de cette catégorie sont les suivants :

### Disponibilité d'un système de développement

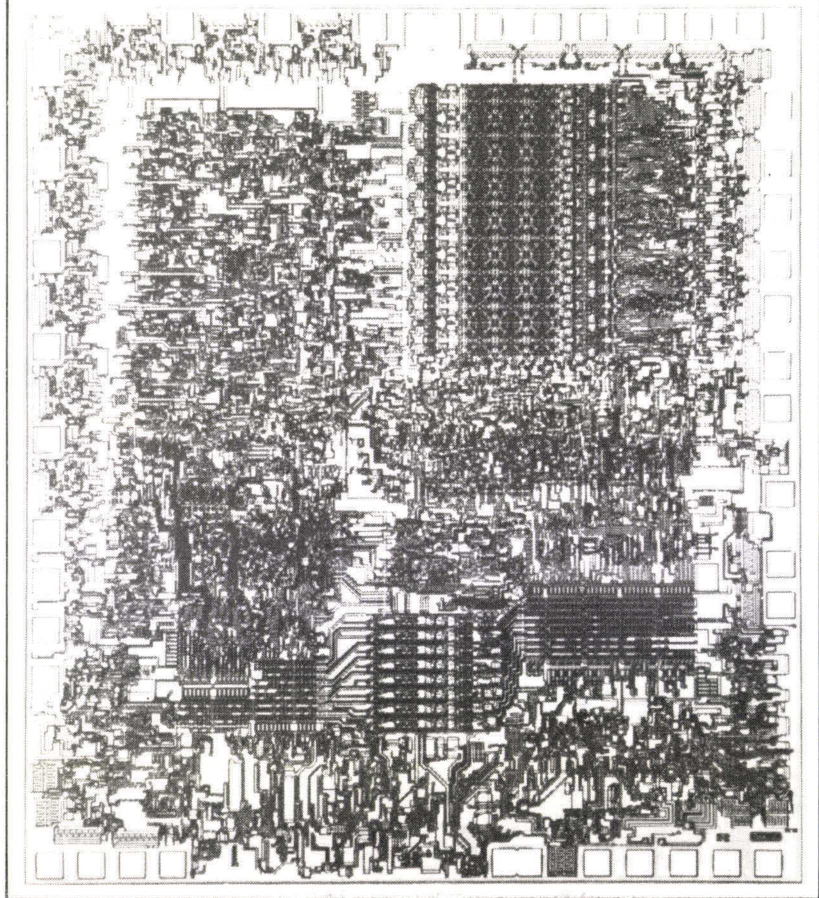
C'est absolument indispensable pour pouvoir réaliser et mettre au point un produit.

Il faut noter que certains systèmes éducatifs peuvent servir de systèmes de développement moyennant un minimum d'extensions. Les microprocesseurs qui ont cela permettent donc à une petite entreprise de démarrer avec un faible investissement.

Sinon un système de développement complet peut coûter jusqu'à 50 000 F. On voit alors l'intérêt d'adopter une famille de microprocesseurs capables de s'adapter à des produits allant du bas de gamme au haut de gamme : l'investissement du système de développement sera à faire une fois pour toutes puisqu'il pourra servir pour tous les produits envisagés.



*Photo 6. – Une unité centrale désormais bien connue : le 8080 d'Intel.*



### Support logiciel

La facilité d'emploi du support logiciel fournit un critère-clé de sélection. Le support logiciel est l'ensemble des programmes qui permettent d'utiliser au mieux le système de développement. On y inclut :

- un moniteur gérant le système. On peut aller jusqu'à un moniteur-disque et un système de gestion de fichiers ;
- un assembleur ;
- un éditeur de textes permettant les corrections.

Il est assez commode que l'assembleur et l'éditeur soient combinés.

- un chargeur. Le mieux est un chargeur-éditeur de liens capable de fournir des programmes relogeables ;
- des compilateurs ou interpréteurs de langages évolués notamment BASIC, APL, PLM (8080), MPL (6800), PLZ (Z-80), CSL 65 (6500) ;
- une bibliothèque de programmes.

Les deux critères précédents sont importants mais ils ne sont pas très discriminatoires : tous les microprocesseurs du marché ont maintenant des systèmes de développement (plus ou moins chers) et un support logiciel assez développé.

## Vue générale du marché des microprocesseurs 8 bits

Donnons maintenant une évaluation rapide des principaux microprocesseurs 8 bits NMOS du marché. Nous nous consacrons à cette gamme car elle est la plus importante du marché (au moins 50 % des applications) et a donné lieu aux produits les plus élaborés. Les paramètres qui varient le plus ici (à défaut d'être les critères les plus décisifs) sont les particularités

hardware, la vitesse, le jeu d'instructions et les possibilités d'adressage.

Nous ne citerons que pour mémoire le SCMP II de National Semiconductors (nettement moins performant que les concurrents) et le F8 de Fairchild (que son jeu d'instructions rébarbatif cantonne dans les applications bas de gamme de sa version monoboîtier, ce qui conduit quoi qu'il en soit à de gros chiffres de ventes).

D'un point de vue historique, les choses se sont passées de la façon suivante. Intel a sorti le premier microprocesseur 8 bits, le 8008 dont l'étude avait été payée par Datapoint qui, en fin de compte, ne l'utilisa pas. Au vu du succès imprévu du produit, Intel et Motorola entreprirent aussitôt d'en préparer des améliorations. Intel sortit le 8080 tandis que Motorola réalisait le 6800 (pas besoin d'autres boîtiers pour constituer une unité centrale alors que le 8080 a besoin d'un 8224 et d'un 8228, tension d'alimentation unique + 5 V alors que le 8080 nécessite aussi du + 12 V et du - 5 V); toutefois, les deux produits ont sensiblement les mêmes performances. Sorti plus tôt, le 8080 a pris une plus large part du marché.

En fait, le 8080 et le 6800 incarnent deux philosophies opposées qui partagent le monde des microprocesseurs 8 bits.

a) Celle du 8080 est principalement caractérisée par :

- cycle machine formé de plusieurs périodes d'horloge ;
- nombreux registres internes ; faibles possibilités d'adressage de la mémoire ;
- entrées-sorties par instructions spéciales.

b) Celle du 6800 implique :

- cycle machine simple formé d'une seule période d'horloge ;
- moins de registres internes mais possibilités de manipulation en mémoire et d'adressage étendues ;
- entrées-sorties projetées en mémoire (ce qui présente des avantages certains).

En outre le 6800 comportait aussi l'idée d'utiliser des boîtiers annexes d'interface efficaces comme le 6820 et le 6850. Intel a été obligé de suivre avec les 8251 et 8255.

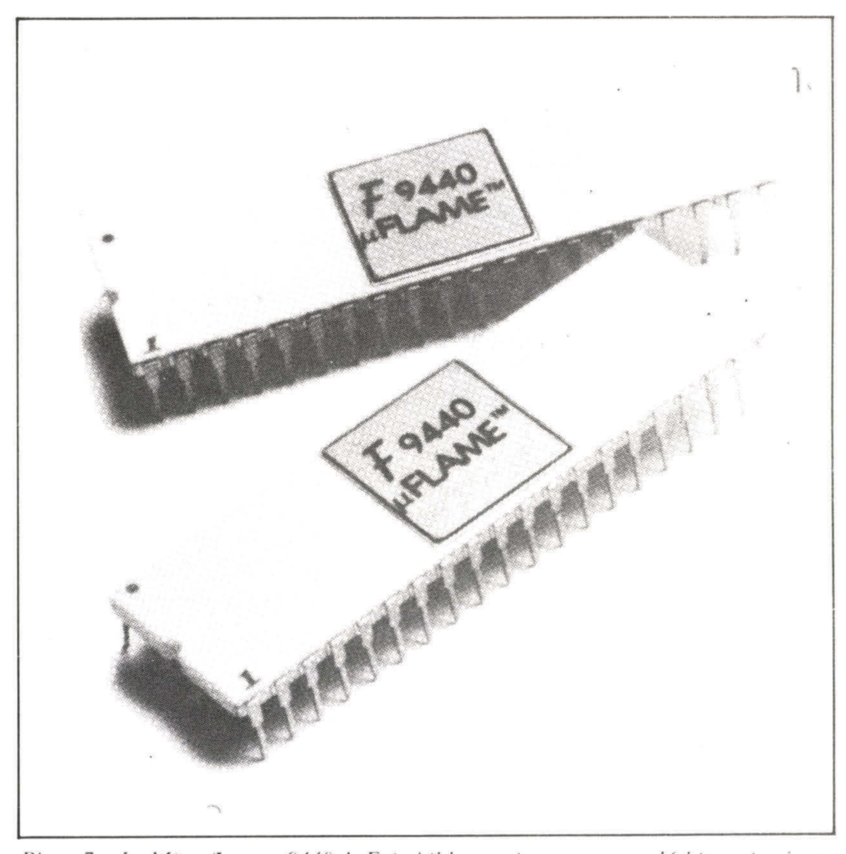


*Photo 7. – Le Microflamme 9440 de Fairchild : un microprocesseur 16 bits qui exécute le jeu d'instructions de la ligne Nova de DATA General.*

Ensuite Signetics a commercialisé un microprocesseur original, extrêmement intéressant, le 2650, qu'on peut considérer comme en quelque sorte intermédiaire entre les deux philosophies. Le 2650 a l'ensemble des modes d'adressage le plus riche du marché (seul le 6500 peut rivaliser avec lui de ce point de vue), mais il est moins performant que ses concurrents (plus lent et son espace adressable plafonne à 32 k) et la philosophie qu'il a adoptée de ne pas avoir de boîtiers d'interface mais d'utiliser des circuits TTL standard est en fait pénalisante (Signetics en est d'ailleurs revenu récemment en annonçant le 2656).

Entre temps, les concepteurs du 8080 quittaient Intel * pour fonder leur propre firme Zilog et sortaient le Z-80 qui est une nette amélioration du 8080. Au point de vue hardware : horloge incorporée, alimentation unique, pas besoin de 8224 ni de 8228, signaux de contrôle plus simples, vitesse d'horloge plus rapide. Au point de vue software, il est compatibles avec le 8080 en ce sens qu'il admet le jeu d'instructions du 8080 comme sous-ensemble du sien de sorte qu'une ROM préparée pour un 8080 peut être utilisée avec un Z-80, mais comporte des améliorations significatives : deux fois plus de registres internes, nombreuses instructions supplémentaires dont certaines sont intéressantes et d'autres font plutôt gadget. Les boîtiers d'interface sont assez attractifs mais chers et aucun n'est multifonctions. En résumé, le Z-80 suit la philosophie du 8080. Il va même plus loin en ce qui concerne le nombre de registres internes, mais fait un pas vers le 6800 en ce sens qu'il possède son adressage indexé. La réponse d'Intel au Z-80 est le 8085 qui a à peu près les mêmes améliorations hardware (elles étaient bien néces-

** Les fondateurs d'Intel sont, eux, des « transfuges » de chez Fairchild.*

saires) mais n'a que deux instructions de plus que le 8080. En revanche, il a une famille de boîtiers annexes intéressants dont certains sont multifonctions.

De la même façon que pour le Z-80, les concepteurs du 6800 quittaient Motorola pour fonder leur propre firme : MOS Technology et produire une amélioration du 6800 : la série 6500. La façon dont les 6500 améliorent le 6800 n'est pas la même que celle du Z-80 par rapport au 8080. Les 6500 sont compatibles hardware avec le 6800 en ce qui concerne les types de cycles et la discipline de bus. Cela signifie que les boîtiers d'interface 6800 peuvent être utilisés avec un 6500 et réciproquement. Il y a cependant quelques améliorations hardware :

a) horloge incorporée,

b) vitesse accrue obtenue de deux manières :

Une architecture améliorée faisant largement appel aux structures pipeline permet de gagner 1 ou 2 cycles d'exécution dans presque toutes les instructions qui sont identiques pour les deux processeurs.

Fréquence d'horloge plus rapide : la fréquence standard est 1 MHz ce qui donne un rendement légèrement amélioré par rapport au 6800 (1 MHz) et des performances du même ordre que celles du Z-80, mais des processeurs à 2 MHz sont disponibles en stock, et des 3-4 MHz peuvent être obtenus sur demande spéciale (plus chers !). Des produits encore plus rapides sont annoncés ce qui donne de loin les meilleures performances du marché.

c) Signaux de contrôle encore plus simples tout en autorisant toutes les possibilités de DMA, etc.

d) Concept de famille : mais l'idée la plus originale de MOS Technology a été de proposer plusieurs microprocesseurs compatibles entre eux. A côté du 6502 (40 broches) qui a toutes les possibilités, figurent des 28 broches (6503, etc.) qui ont un espace adressable réduit (4 k ou 8 k) et qui diffèrent par les signaux de contrôle. Ceci est particulièrement intéressant pour constituer des systèmes minimaux. Il est possible de choisir le membre de la famille le mieux adapté à l'application, ce qui permet de concevoir des systèmes minimaux et des systèmes de hautes performances avec un 3 MHz.

Au point de vue software les 6500 ont un jeu d'instructions qui suit une philosophie très semblable au 6800 : les entrées-sorties sont projetées en mémoire. On va même plus loin au point de vue modes d'adressage puisqu'on dispose en plus de l'adressage absolu indexé, de l'adressage indirect préindexé et post-indexé. Seul le 2650 peut rivaliser avec ces modes d'adressage.

Les boîtiers d'interface associés sont aussi parmi les plus intéressants du marché. Il y a notamment des boîtiers multifonctions ROM, RAM, Ports d'E/S, temporisateur (6530), un circuit RAM, Ports d'E/S, temporisateur (6532) et enfin un boîtier ports d'E/S parallèles avec handshaking, registre série, deux temporisateurs sur 16 bits (6522).

Tout ceci explique qu'une statistique pour l'année 1978 fournie par la société américaine Dataquest, indépendante de tout constructeur, place la famille 6500 en tête des ventes de sa catégorie. ■

**D.-J. David**

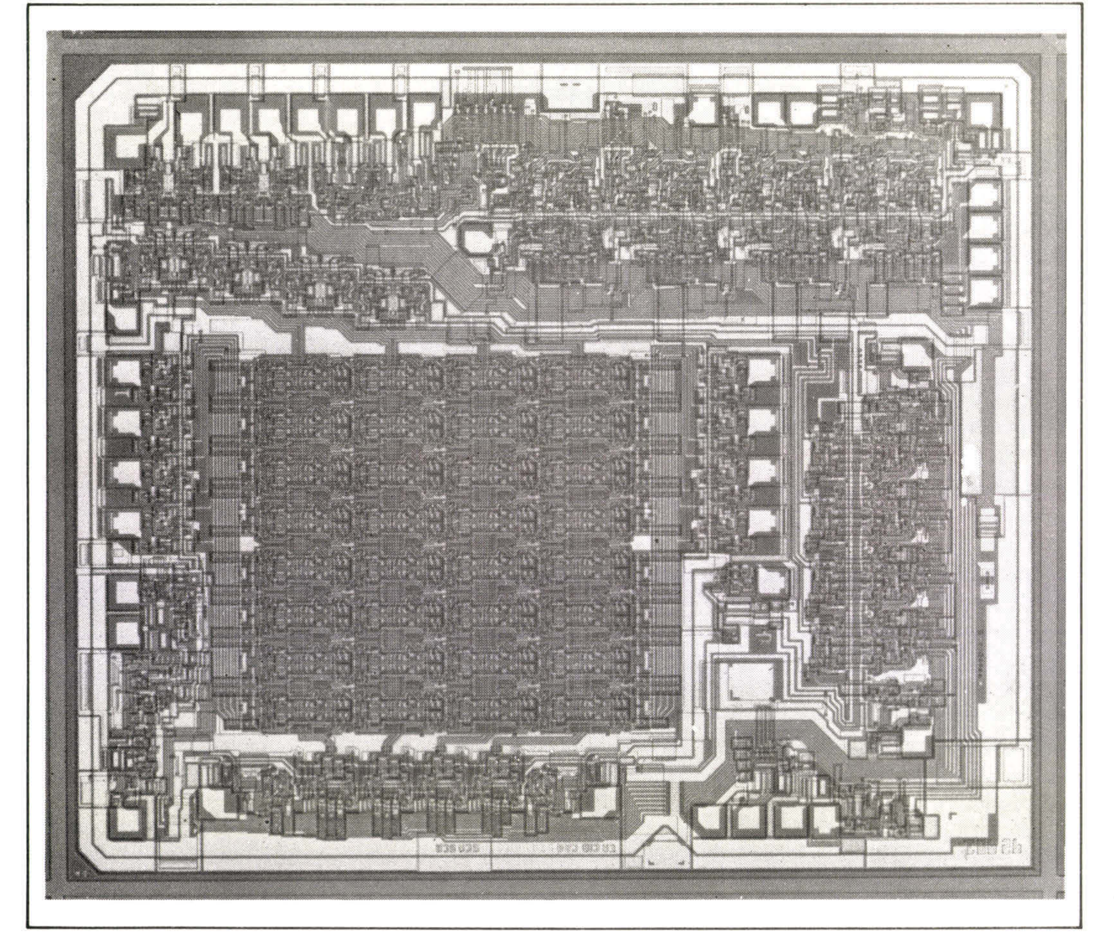
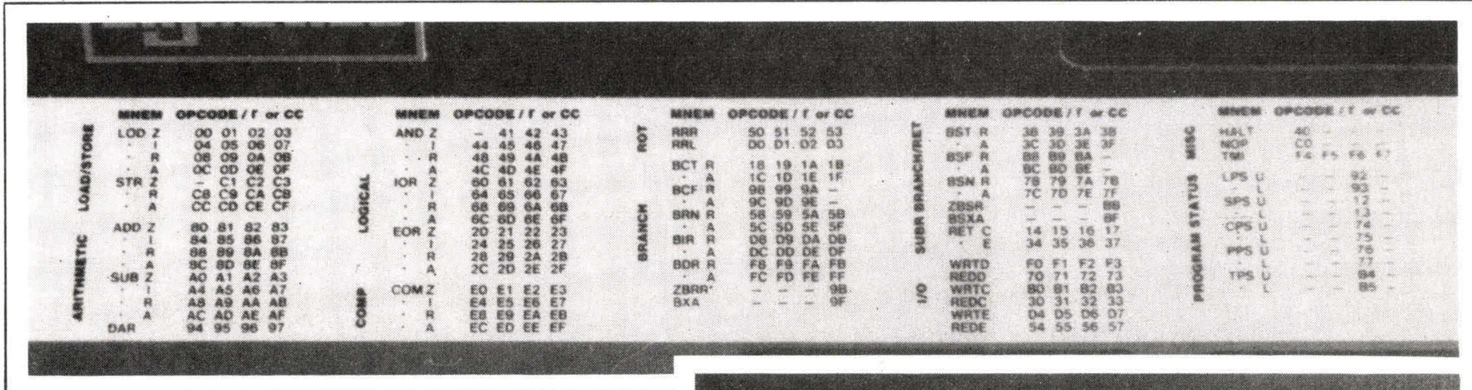
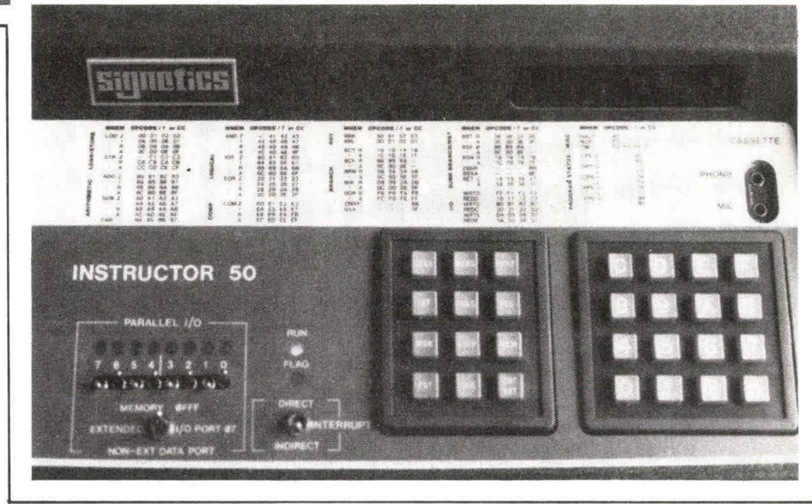


*Photo 8. – Réalisé en technologie bipolaire, le SF.C 92901 est un microprocesseur rapide qui traite des mots de 4 bits (cliché Thomson-CSF).*

# Six leçons pour programmer

**Une programmation structurée ou l'organisation des programmes**




*Sur la nouvelle version de l'Instructor 50 de Signetics, le tableau du jeu d'instructions du microprocesseur 2650 est sérigraphié sur la face avant.*

**Dans le précédent article de notre série, nous nous sommes attachés à décrire plus particulièrement, dans le répertoire d'instructions de notre microprocesseur 2650, celles qui traitent ou manipulent l'information : recopies, additions, opérations logiques, etc. , ainsi que les différents formats de ces instructions.**

**Ces instructions de traitement sont, en elles-mêmes, insuffisantes à remplir toutes les fonctions d'un ordinateur, et notamment celles qui donnent à ces fameuses machines une soi-disant « intelligence » , à savoir les fonctions de décision.**

**Souvenons-nous de notre introduction intuitive du bit, comme de la plus petite quantité d'information pratique, permettant de déterminer un choix élémentaire entre deux options.**

**De manière analogue, nous sommes conduits à étudier aujourd'hui la structure de décision la plus élémentaire, autrement dit : l'alternative.**

## Si... alors... sinon

En bon français, nous exprimons souvent nos choix à l'aide des trois **mots-clés** « si », « alors » et « sinon » : « **SI** je n'ai plus beaucoup d'essence **ALORS** je m'arrête à la prochaine station-service **SINON** je poursuis ma route. » Très souvent, l'alternative « sinon » consiste à « ne rien faire », autrement dit, à « poursuivre », et l'on s'exprime plus simplement : « Si je n'ai plus beaucoup d'essence alors je m'arrête à la prochaine station-service » ; point à la ligne !

En examinant cette manière typique de s'exprimer, on verra sans peine que sa forme la plus générale est :

« **SI** condition **ALORS** action si la condition est **vraie SINON** action si la condition est **fausse.** »

Appliquée au micro-ordinateur, cette première structure de décision/action se traduira nécessairement par les étapes suivantes :

- **Elaboration de la condition :** par exemple, la comparaison de deux nombres demande un certain « calcul ».
- **Exécution** de certaines **instructions** si la condition est **vraie.**
- **Exécution** d'**autres instructions** si la condition est **fausse.**

## Code condition et comparaisons

En termes d'ordinateur, il n'y a guère que deux sortes de conditions simples qui puissent faire l'objet de **tests** élémentaires :

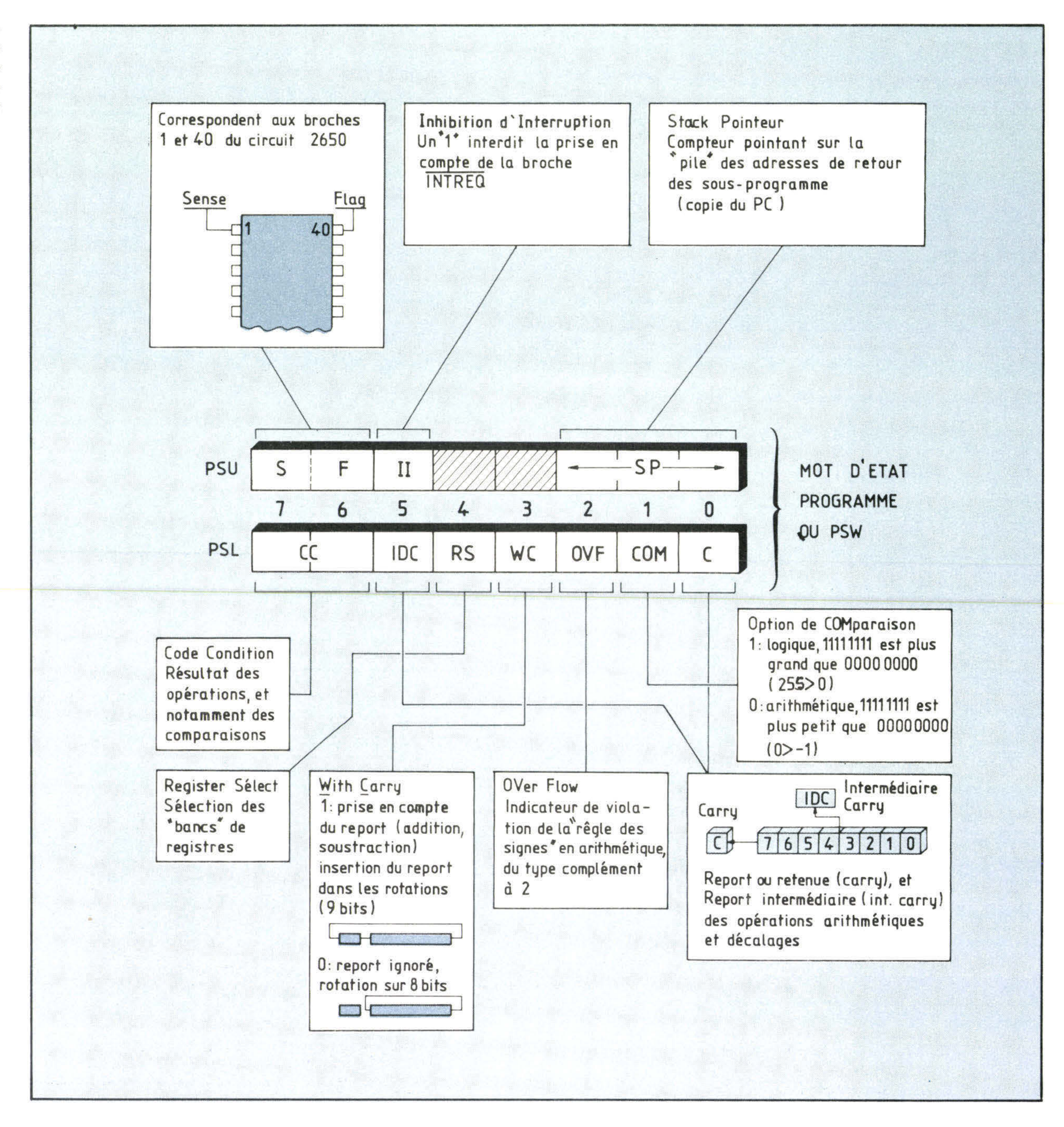
- les valeurs **comparées** de deux nombres, menant à l'un ou l'autre de trois résultats : « plus petit », « égal », « plus grand » ;
- les valeurs **binaires** pures, égales à « 1 » (souvent interprétées comme « vrai », par convention), ou à « 0 » (« faux », dans la même convention).

Pour le microprocesseur 2650, ceci se matérialise par deux bits du **mot d'état** (voir définition dans notre encadré) ; plus précisément, les deux bits de gauche du PSL forment le **Code-Condition, CC** en abrégé.

**Toutes** les instructions qui **affectent** un registre, outre le fait qu'elles déposent dans ce registre un « résultat », ont pour effet de « positionner » le CC en fonction d'une règle très simple : selon le « **signe** » du résultat.

Toutes les instructions qui affectent un registre positionnent les bits du code condition (CC) dans le registre d'état.

*Définition de chacun des bits du mot d'état du microprocesseur 2650. PSU est le mot de poids fort et PSL le mot de poids faible.*




| Résultat | CC |
|---|---|
| < 0 | 10 |
| = 0 | 00 |
| > 0 | 01 |

Pour cette notion de « signe », la conversion est liée à la notion de nombre en complément à deux. On pourrait dire aussi que chaque résultat est automatiquement « comparé à zéro ».

Cela dit, il y a beaucoup de ces pratiques où nous souhaiterons **comparer** deux octets **sans autre effet** que l'affectation du CC. Pour ce faire, on a prévu dans le répertoire un jeu d'instructions spécial, résumé comme suit.

| Instruction | 1er terme | 2e terme |
|---|---|---|
| COM Z | Reg. 0 | Autre registre |
| COM I | Registre | Valeur immédiate |
| COM R | Registre | Mémoire (adresse relative) |
| COM A | Registre | Mémoire (adresse absolue) |

Lorsque l'on exécute une telle instruction, registre et/ou mémoire restent intactes ; seul le CC est affecté :

| 1^er^ terme : : 2^e^ terme | CC |
|---|---|
| < | 1 0 |
| = | 0 0 |
| > | 0 1 |

Ceci est bien cohérent avec la règle précédente. En effet, pour ce faire, le microprocesseur effectue une **soustraction :** il retranche le second terme du premier et le résultat affecte le CC, mais n'est, par ailleurs, recopié nulle part...

Il reste à lever une ambiguïté. Si l'on « compare » les deux octets 1111 1111 et 0000 0000, est-ce que :
— le premier est plus grand que le second (255 > 0), en valeur « naturelle »,
— ou plus petit (— 1 < 0) dans l'interprétation « avec signe » du complément à deux ?

Pour lever le doute, le 2650 prévoit que l'on puisse, par programme, prendre l'**option** souhaitée ; techniquement, cela se fait grâce au positionnement (instruction PPSL) du bit COM du PSL, qui est exploité comme suit :

| COM = | Comparaison | Exemple |
|---|---|---|
| 1 | Logique (« naturelle ») | 1111 1111 > 0000 0000 |
| 0 | Arithmétique (« signée ») | 1111 1111 < 0000 0000 |

## Test de bits

Il est fréquent, dans les applications, d'avoir à considérer (par exemple : parce qu'ils représentent des contacts) un ou plusieurs bits d'un octet et de décider d'une action selon qu'ils sont ou non positionnés à « 1 ».

Pour ce test, par nature différent des comparaisons numériques déjà évoquées, le 2650 prévoit quelques instructions particulières.

| TMI | Test de certains bits d'un registre |
|---|---|
| TPS U | " " du PSU |
| TPS L | " " du PSL |

Ces instructions sont sur deux octets, dont le second constitue un **« masque »** qui désigne, par des « 1 », quels bits sont **testés** et par des « 0 », ceux qui ne sont pas considérés. Ainsi l'instruction :

| Registre 1 | Registre 2 |
|---|---|
| 1111 0101 | 0100 1000 |
| 7 0 | 7 0 |

considèrera les bits de rang 3 et 6 du registre 1. Dans ce cas, le CC est affecté différemment :

| | |
|---|---|
| **Tous** les bits considérés sont à « 1 » | CC = 00 |
| **Un** des bits considérés (au moins) est à « 0 » | CC = 10 |

## Les sauts conditionnels

Voilà pour le SI... Il nous reste à montrer comment s'interprètent, dans le microprocesseur, le ALORS... et le SINON... de notre alternative ; nous avons déjà dit qu'ils ne pouvaient se traduire que par l'exécution de deux **séquences d'instructions distinctes,** en fonction de la condition.

Si nous devions « inventer » l'instruction « ad hoc », nous verrions rapidement qu'il est commode que **l'une des séquences** soit... en séquence ; autrement dit, dans l'un des deux cas, nous laisserions le programme se poursuivre « naturellement » avec la progression du compteur ordinal (PC).

Pour le cas contraire, il nous faudrait (par définition) opérer une **rupture de séquence,** qui ne peut se traduire techniquement que par le **forçage** dans le compteur ordinal, de l'**adresse de l'autre séquence ;** ou plus précisément, de l'adresse de la première instruction qui traite l' « autre cas ».

On trouve précisément, dans le répertoire, une série d'instructions de ce genre, qui ont l'allure suivante :

« BCT » XX adresse $\alpha$

et s'interprètent comme suit :

**SI** (le code-condition CC coïncide avec le code XX de l'instruction)
**ALORS** (on va exécuter l'instruction suivante à l'adresse $\alpha$, qui est « recopiée » dans le compteur-ordinal)
**SINON** (le compteur ordinal progresse normalement, et l'on exécute l'instruction suivante).

Pour les programmeurs, ce type d'instruction s'appelle un **saut** (angl. : jump) ou aussi un **branchement** (angl. : branch). D'où, très naturellement pour des concepteurs parlant anglais, des mnémoniques avec le préfixe **BCT** ou : « **B**ranch on **C**ondition **T**rue » (Branchement sur Condition Vraie).

Il est extrêmement commode d'avoir aussi des branchements sur **condition fausse,** afin d'exprimer, notamment, ce que les mathématiciens appellent des inégalités larges, ou bien la condition « différent » :

| Cette condition est VRAIE | Si cette condition est FAUSSE |
|---|---|
| $a \geqslant b$ | $a < b$ |
| $a \neq b$ | $a = b$ |
| $a \leqslant b$ | $a > b$ |

Pour ce faire, les concepteurs du 2650 ont prévu d'autres code-opératoires, dont le préfixe est **BCF** ou : **B**ranch on **C**ondition **F**alse (... Fausse).

Quant aux adresses, elles peuvent être absolues (suffixe **A**) ou relatives (préfixe **R**).

Ainsi, nous pouvons définir un

Une rupture de séquence se traduit par forçage, dans le compteur ordinal, de l'adresse d'une autre séquence.

* *Six leçons pour programmer : Micro-Systèmes nº 3, page 95.*

jeu de quatre instructions :
– BCT R
– BCT A
– BCF R
– BCF A
qui, combinées avec les trois valeurs du CC, permettent d'exprimer **toutes** les décisions usuelles sur comparaison.

Ceci est expliqué dans le tableau des codes-op * ; nous avons disposé autant de colonnes que de « cas » de sauts après comparaisons des différents types :

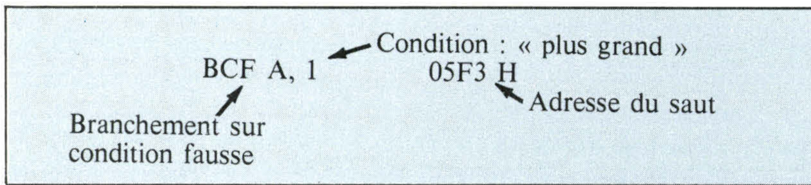
$$< \leqslant = \geqslant > \neq$$

Chaque case donnant la traduction correcte, en hexadécimal.

Ainsi le « branchement, si $\leqslant$, à l'adresse hexa 05F3 » se traduit comme

| | | |
|---|---|---|
| 9 D | 0 5 | F 3 |

ce qui s'exprime sous la forme symbolique




Vous avez probablement remarqué que le CC ne peut avoir que trois valeurs : 00, 01 et 10. Et 11 ? Eh bien, pour le 2650, cette combinaison s'interprètera comme une condition « toujours vraie ». Autrement dit, les instructions BCT, avec le code 11, provoqueront **toujours** un saut à l'adresse donnée. Ce qui, pour un programmeur, s'appelle un **saut** (branchement) **inconditionnel.**

Quant à l'alternative, nous l'exprimons à l'aide d'une notation, très simple, valable **pour n'importe quel ordinateur !** En premier lieu, la **condition** est « testée » (par calcul + saut conditionnel) ; la flèche « faux » exprime la rupture de séquence dans l'un des deux cas testés ; si la condition est vraie, on exécute « ceci » en séquence **et l'on se déroute** vers « après » **par un saut inconditionnel.** A la suite (en mémoire) sont disposées les instructions qui réalisent « cela »... et les deux déroulements **se concluent au même point (fig. 1 b).**

Pourquoi cette disposition de l'alternative ? (il en existe une autre tout à fait équivalente, que vous trouverez sans peine). Pour la bonne raison que les instructions ainsi groupées peuvent, **en bloc, s'insérer dans une autre séquence ;** ce qui est rendu possible par l'observation de la règle ci-dessous : la « structure » n'a qu'**une seule entrée** (« avant ») et **une seule sortie** (« après »).

Nous pouvons signaler tout de suite les deux variantes immédiates de ces structures représentées **figure 2 a** et **2 b** qui se comprennent d'elles-mêmes avec nos notations élémentaires.

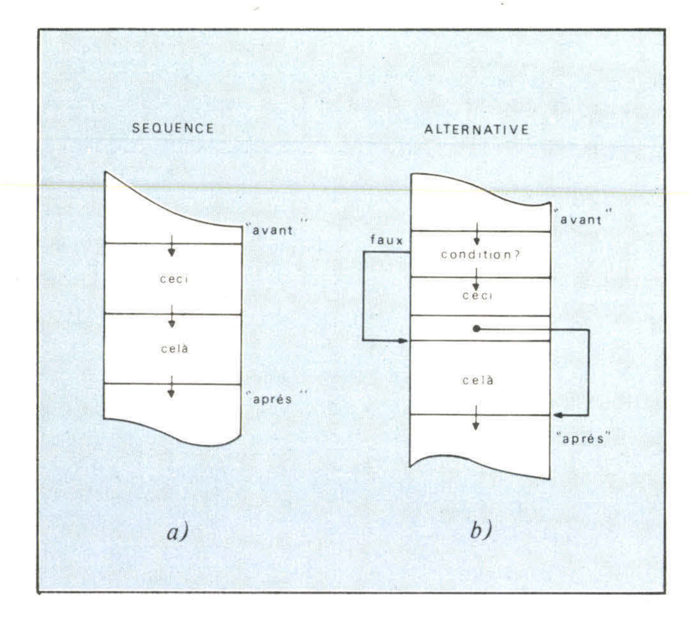



*Fig. 1. –*
*a) Sequence pure : après avoir exécuté « ceci », on exécute « cela ».*
*b) Alternative : Si la condition est vraie, on exécute « ceci » ; si elle est fausse, on exécute « cela ». Dans un cas comme dans l'autre, on exécute ensuite « après ».*

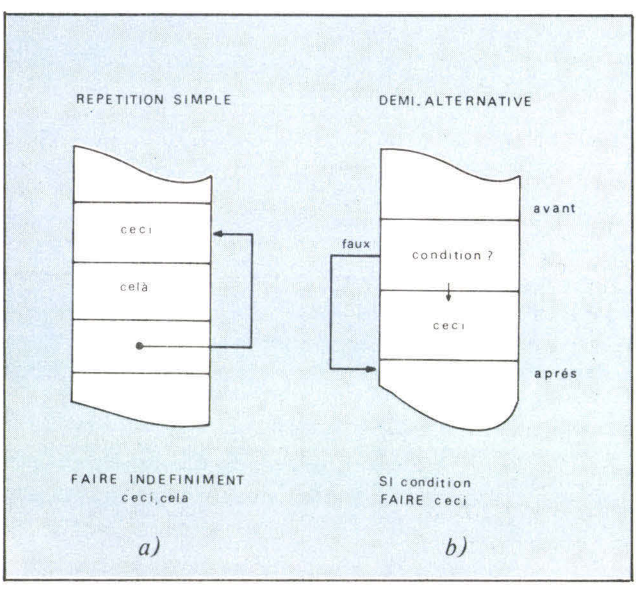



*Fig. 2.*
*a) Répétition simple : le programme fait indéfiniment « ceci » – « cela ».*
*b) Demi alternative : si la condition est vraie ou exécute « ceci » sinon « après ».*

## Premières structures de programme

Avec les sauts des types étudiés, nous pouvons commenter les deux **structures** de programme les plus élémentaires :

- la **séquence pure :** FAIRE ceci (puis), FAIRE cela, etc.
- l'**alternative** que nous avons déjà examinée : SI (condition) ALORS (FAIRE ceci) SINON (FAIRE cela).

Leur **traduction** à l'aide d'instructions implantées dans une mémoire peut être littéralement « normalisée » selon deux **schémas standards** donnés **fig. 1 a** et **1 b.**

La pure séquence ne pose aucun problème : on écrit les instructions qui font « ceci », puis (dans l'ordre de la mémoire) les instructions qui font « cela » **(fig. 1 a).**

## Un exemple : analyse...

Abstrait, ce qui précède ? Au contraire ! Nous allons voir sur un exemple très simple comment notre exposé s'exploite pour donner une solution **directe** à un problème, simplement énoncé en bon français ! Et ce, à condition d'aller **du général au particulier ;** ne sommes-nous pas au pays de Descartes ?

Imaginons que notre premier énoncé du problème soit :

*Simuler un codeur activé par pres-*

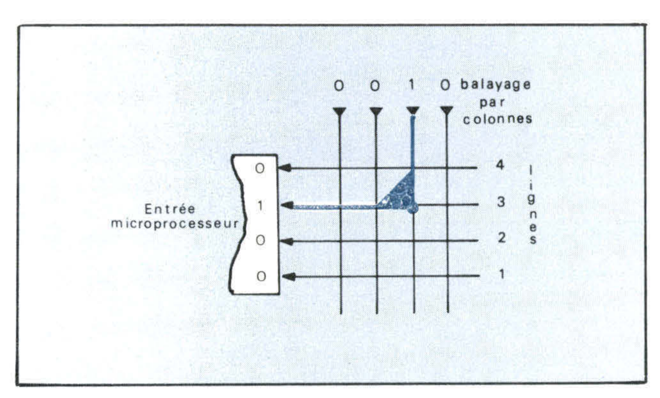



*Fig. 3. – Principe du balayage par colonne : un niveau 1 est successivement appliqué à chaque colonne. Si une touche est enfoncée, ce niveau 1 est transmis sur la ligne correspondante. Le numéro de la ligne où a lieu le contact peut donc être déterminé par le microprocesseur.*

*sions de la boucle SENSE de l'instructeur, qui allume les voyants du pupitre en fonction des clés, selon le code suivant :*

| Clés | Valeur sur les voyants |
|---|---|
| XXXX 0000 | 0 |
| XXXX 0001 | 1 |
| XXXX 0010 | 2 |
| XXXX 0100 | 3 |
| XXXX 1000 | 4 |
| ... autre... | inchangés |

Pour votre curiosité, cette opération a un sens, dans le codage de claviers explorés par balayage **(fig. 3)** et où l'on admet qu'une touche **au plus** doit être enfoncée dans une « colonne » (problème de la double frappe). Le codage consiste à donner le numéro de la « ligne » où a lieu le contact.

Notre projet de programme, dans une première **analyse,** peut s'énoncer comme suit :

**FAIRE INDEFINIMENT :**
- **SI** SENSE = 1 **ALORS** CODER LE RESULTAT
- AFFICHER LE RESULTAT

et donne le schéma de programme de la **figure 4.**

Nous voyons intuitivement que le bloc baptisé « CODER » n'est **pas encore** décomposé en parties assez élémentaires pour que leur programmation soit « évidente ». A l'aide du tableau de codage, nous pouvons le reformuler comme suit :

**CODER :**
- LIRE les clés
- **SI** (clés = 0) **ALORS** (résultat — 0)
- **SI** (clés = 1) **ALORS** (résultat — 1)
- **SI** (clés = 2) **ALORS** (résultat — 2)
- **SI** (clés = 4) **ALORS** (résultat — 3)
- **SI** (clés = 8) **ALORS** (résultat — 4)

Ce qui donne immédiatement le schéma **(fig. 5)** du programme dans lequel nous vous laisserons poursuivre de vous-même, d'après le modèle de traduction des deux premiers « SI... ».

## ... Et programmation

Pour écrire sur l'I 50 un programme qui traduit exactement les schémas dégagés ci-dessus, il n'y a plus qu'à régler quelques petits détails techniques :
- on prend (arbitrairement) le registre 1 comme « résultat » ;
- on choisit de lire/écrire sur les clés/voyants dans le mode « non-étendu » ;
- on se souvient que SENSE apparaît comme le bit 7 du pseudo-registre PSU, et le « teste » par l'instruction... TPS U.

Dans ces conditions, on écrit **quasi automatiquement** le programme EXEMPLE 4.1 de la **figure 6** où l'exercice, pour le lecteur, consiste à **retrouver la structure** dans le texte du programme.

Vous imaginerez probablement de « meilleurs » programmes pour faire le même travail. Posez-vous alors la question : meilleurs, peut-être, mais à quel titre ?

La **place** occupée ? Nous doutons que vous gagniez mieux que 15 % (l'usage de branchements **relatifs** n'apporte aucune amélioration **de principe ;** nous savons « gagner » à cet égard les mêmes 7 octets que n'importe qui).

Le **temps d'exécution ?** Il est possible de le diminuer de quelques dizaines de microsecondes. La question suivante est alors très méchante : pour quoi faire ?

La **clarté,** la **sûreté** du programme ? Voilà **le** point sur lequel nous nous sentons à peu près inattaquables !

## (Bref) discours sur la méthode *

Pour en convaincre le lecteur, nous lui livrons en pâture une autre version du même programme, tel qu'à coup sûr un programmeur « novice et indiscipliné » serait fier de l'avoir écrit. Nous vous proposons à titre d'exercice (pas facile du tout) de :

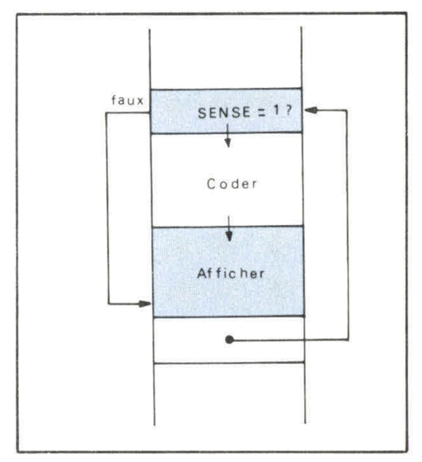
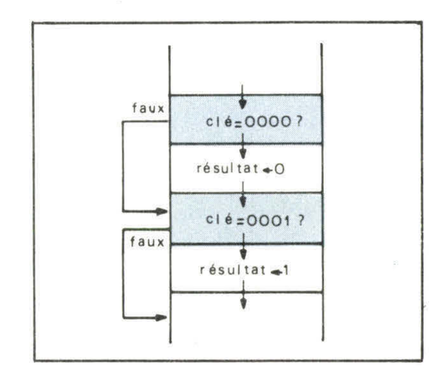



*Fig. 4. – Faire indéfiniment :*
*Si SENSE = 1, alors coder le résultat puis l'afficher.*
*Si SENSE = 0, afficher directement le résultat.*

*Fig. 5. – Début du schéma du programme de codage. Il vous appartient de le poursuivre de vous-même d'après le modèle de traduction des deux premiers SI.*




* *Pardon, Monsieur Descartes !*

De nombreux traitements de l'information se traduisent par la répétition d'une séquence d'instruction jusqu'à l'obtention du résultat.

**Programme** : EXEMPLE 4.1 **Auteur** : J.-M. COUR **Date** : 01/79 **Page** : 1/1

| Adresse | Traduction | Etiquette | Opération | Paramètres | Commentaires |
|---|---|---|---|---|---|
| 000 | B4 80 | DEBUT | TPSU | SENSE | TEST BIT 7 PSU |
| 002 | 9C 00 2B | | BCFA, 0 | AFFICHER | SAUT SI ≠ 1 (CC ≠ 0) |
| 005 | 70 | CODER | REDD | R0 | R0 — clés (LIRE) |
| 006 | 44 0F | | ANDI, R0 | 0000 1111 | GARDER BITS 3-0 |
| 008 | E4 00 | TEST 0 | COMI, R0 | 0 | COMPARER AVEC 0000 |
| 00A | 9C 00 0F | | BCFA, 0 | TEST 1 | SAUT SI CC ≠ 0 |
| 00D | 05 00 | | LODI, R1 | 0 | SINON, R1 — 0 |
| 00F | E4 01 | TEST 1 | COMI, R0 | 1 | TEST CLES = 0001, |
| 011 | 9C 00 16 | | BCFA, 0 | TEST 2 | ETC. |
| 014 | 05 01 | | LODI, R1 | 1 | |
| 016 | E4 02 | TEST 2 | COMI, R0 | 2 | |
| 018 | 9C 00 1D | | BCFA, 0 | TEST 4 | |
| 01B | 05 02 | | LODI, R1 | 2 | |
| 01D | E4 04 | TEST 4 | COMI, R0 | 4 | |
| 01F | 9C 00 24 | | BCFA, 0 | TEST 8 | |
| 022 | 05 03 | | LODI, R1 | 3 | |
| 024 | E4 08 | TEST 8 | COMI, R0 | 8 | |
| 026 | 9C 00 2B | | BCFA, 0 | AFFICHER | |
| 029 | 05 04 | | LODI, R1 | 4 | |
| 02B | F1 | AFFICHER | WRTD, R1 | | VOYANTS — (R1) |
| 02C | 1F 00 00 | | BCTA, 3 | DEBUT | SAUT INCOND. DEBUT (FAIRE INDEFINIMENT) |

*Fig. 6. – Vous pouvez essayer, à titre d'exercices, de modifier ces deux programmes pour y introduire de nouvelles spécifications (tels que modifier le nombre de lignes à lire du clavier ou changer le code).*

**Programme** : EXEMPLE 4.2 **Auteur** : J.-M. COUR **Date** : 01/79 **Page** : 1/1

| Adresse | Traduction | Etiquette | Opération | Paramètres | Commentaires |
|---|---|---|---|---|---|
| 000 | B4 80 | DEBUT | TPSU | SENSE | |
| 002 | 9C 00 00 | | BCFA, 0 | DEBUT | |
| 005 | 70 | | REDD | R0 | |
| 006 | 1C 00 21 | | BCTA, 0 | SUITE | |
| 009 | E4 01 | | COMI, R0 | 1 | |
| 00B | 1C 00 21 | | BCTA, 0 | SUITE | |
| 00E | E4 02 | | COMI, R0 | 2 | |
| 010 | 1C 00 21 | | BCTA, 0 | SUITE | |
| 013 | E4 04 | | COMI, R0 | 4 | |
| 015 | 9C 00 1A | | BCFA, 0 | TEST 8 | |
| 018 | 04 03 | | LODI, R0 | 3 | |
| 01A | E4 08 | TEST 8 | COMI, R0 | 8 | |
| 01C | 9C 00 21 | | BCFA, 0 | SUITE | |
| 01F | 04 04 | | LODI, R0 | 4 | |
| 021 | F0 | SUITE | WRTD, R0 | | |
| 022 | 1F 00 00 | | BCTA, 3 | DEBUT | |

- trouver comment il marche : le programmeur (novice et indiscipliné) a jugé superflu de le commenter...
- trouver (par l'expérience ou le raisonnement) qu'en fait il ne marche pas bien : seul quelques « bons cas » vont « passer »...
- essayer de **le corriger sans le refaire !**

Pour mesurer davantage l'étendue des dégâts, essayez de **modifier les deux programmes** EXEMPLE 4.1 **(fig. 6)** et EXEMPLE 4.2 pour y introduire une nouvelle spécification. Nous l'inventons pour les besoins de la cause, mais elle ressemble assez aux « petites modifications » (!) qu'un client « naïf » demande tout naturellement :

« On lit non plus 4, mais 5 lignes du clavier, et le code est comme suit. »

| Clés | Voyants (déc.) |
|---|---|
| XXX 0000 0 | 0 |
| XXX 0000 1 | 1 |
| XXX 0001 0 | 2 |
| XXX 0010 0 | 3 |
| XXX 0100 0 | 4 |
| XXX 1000 0 | 5 |

## Les répétitions : tant que... jusqu'à...

Beaucoup de traitements de l'information se traduisent par la répétition d'une séquence d'instructions **jusqu'à** l'obtention d'un résultat, ou **tant qu'**une certaine condition est réalisée.

Pour ces deux expressions bien françaises aussi, il est possible d'établir des schémas de programme équivalents et d'une grande simplicité pour les phrases : « **TANT QUE** telle condition **FAIRE** ceci »
et
« **FAIRE** cela **JUSQU'A** telle condition ».

Les schémas des **figures 7 a** et **7 b ne sont pas équivalents,** car dans la structure **JUSQU'A** l'opération « cela » s'effectue **au moins une fois** avant que l'on teste si l'on

a atteint l'**objectif** (exprimé par la « condition »).

Une variante importante de ces schémas s'exprime comme suit :
« **FAIRE** ceci, **N FOIS** »
et consiste en la répétition du traitement « ceci », un nombre déterminé de fois (N). On vérifie facilement qu'une expression équivalente est :

- Armer un compteur avec la valeur N
- **FAIRE** « ceci » **JUSQU'A** ce que la soustraction de 1 au compteur donne zéro.

Le schéma de ce programme est donné **figure 8.**

Ce schéma est tellement fréquent que l'on trouve dans le répertoire des microprocesseurs des instructions spécialement étudiées pour en rendre compte. Pour le 2650, quatre instructions sont là :

BDR R    BIR R
BDR A    BIR A

qui se lisent comme **B**ranchements, sur **D**écrémentation ou **I**ncrémentation * d'un **R**egistre, avec les habituelles variantes d'adressage **R**elatif ou **A**bsolu.

Un usage **très important** de ces derniers schémas est la pure **temporisation** qui consiste en ceci :
« **FAIRE** rien du tout **N FOIS** »
ce qui a l'air curieux, sauf si l'on se souvient qu'aucune opération dans un ordinateur n'est instantanée, s'agissant d'instructions dont le temps est déterminé avec une haute précision par un quartz, le **délai** sera prévisible avec la même précision !

Dans un cas très simple, le couple d'instructions

| | | | |
|---|---|---|---|
| | LODI | R3, x | INIT. COMPTEUR R3 |
| ICI | BDRA, R3 | ICI | DECOMPTE JUSQU'A (R3) = 0 |

réalise une temporisation de 2 cycles (LODI), plus x fois les 3 cycles de BDRA ; d'où un temps de (3 x + 2) cycles.

La valeur du cycle dépend bien sûr du quartz d'horloge. Attention, voyez bien que x = 0 provoque 256 tours !

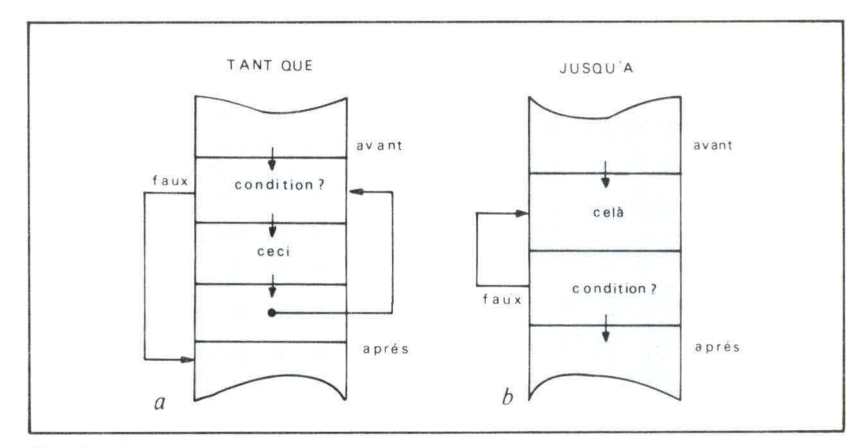



*Fig. 7. – Exemple de répétition :*
*a) Faire « ceci » tant que la condition est vraie.*
*b) Faire « cela » jusqu'à ce que la condition soit vraie.*

Il est extrêmement utile de disposer de séquences de temporisation-type ; aussi en donnons-nous deux pratiques pour le 2650, et qui utilisent le format **relatif :** en effet (par nature) ceci autorise la recopie de telles instructions... n'importe où dans un programme. Les registres R2 et R3 sont pris à titre d'exemple :

| Temporisation « courte » | Temporisation « longue » |
|---|---|
| **Symbolique**<br>LODI R3, X<br>ICI BDRR R3, ICI | **Symbolique**<br>LODI R3, X<br>LODI R2, Y<br>ICI BDRR, R3 ICI<br>BDRR, R2 ICI |
| **Hexadécimal**<br>07 xx<br>FB 7E | **Hexadécimal**<br>07 xx<br>06 yy<br>FB 7E<br>FA 7C |
| **Formule**<br>t = 3 X + 2 | t = 771 Y + 3 X – 764 |

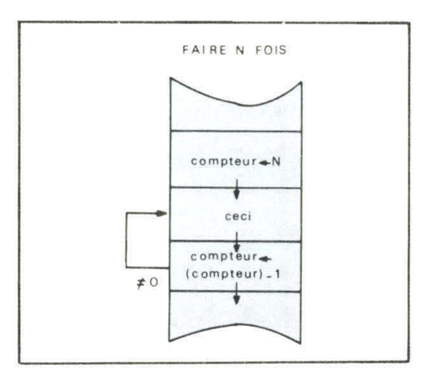



*Fig. 8. – Un compteur est chargé avec la valeur N. Puis le programme fait « ceci » jusqu'à ce que les soustractions successives de 1 du compteur donnent zéro. Ce schéma, très utilisé, peut être par exemple celui d'une boucle de temporisation.*

** Noms poétiques donnés par les informaticiens aux opérations toute bêtes qui consistent à ajouter 1 (incrémenter) ou retrancher 1 (décrémenter). Restons simple !*

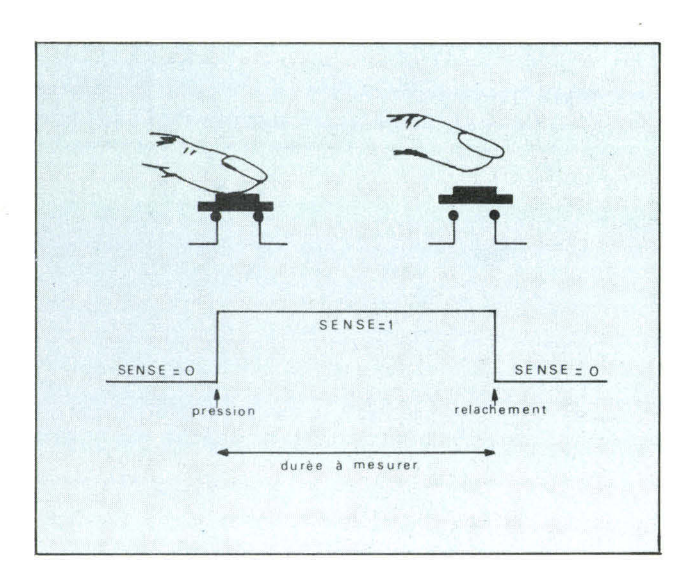



*Fig. 9. - Soit à mesurer le temps qui s'écoule durant l'impulsion créée en pressant sur la touche SENSE...*

**Exercice**

Retrouvez vous-même le « pourquoi » de la seconde formule ; ce n'est pas si évident ! Rappelons que le cycle (sur l'I 50) est de 3,33 microsecondes : calculer les délais maxima pour X et Y = 256 (constantes xx = yy = 00 !)

## Exemple : « mesure » d'une impulsion

Posons-nous comme problème de **mesurer,** à l'aide d'un programme « rudimentaire », le temps d'une **impulsion,** que nous créerons en pressant la touche SENSE **(fig. 9).**

Une analyse succincte nous conduit à décrire la structure du programme qui suit :

**DEBUT :**
- Mesure ← 0
- **Jusqu'à** SENSE = 1, ne rien faire.
- **Jusqu'à** SENSE = 0, FAIRE :
  - Attendre « un certain temps * »
  - Mesure ← Mesure + 1
- Afficher la mesure.
- Halte.

** On compte des attentes.*

Le programme EXEMPLE 4.3 **(fig. 10)** est une traduction qui obéit aux schémas de programme-type donnés précédemment pour JUSQU'A. Délibérément, les constantes de comptage X et Y sont laissées à votre discrétion pour que vous fassiez varier la **précision** (meilleure si la temporisation est courte) et la **valeur maximale mesurable,** d'autant plus grande que la temporisation est longue. Cherchez à optimiser les compromis... et à quel titre ils sont « meilleurs » !

## Où en sommes-nous ?

Nous nous sommes proposés dans cette série, répétons-le, d'apporter au lecteur (sous une forme aussi simple que possible) **des bases très solides,** sur lesquelles on peut bâtir de « bons » logiciels. Notre propos n'est pas d'épuiser le sujet, mais plutôt de donner de bonnes **orientations** pour l'effort personnel de chacun ; et par là même, d'éviter de nombreux pièges, hélas, classiques !

Dans le présent numéro, nous avons donné assez d'indications pour qu'avec un peu de réflexion et d'expérience, le lecteur puisse constater :
- qu'en y réfléchissant bien, **tout traitement** de l'information peut être spécifié par combinaison de **phrases-type** énoncées dans la langue de tous les jours ;
- qu'à ces différentes phrases correspondent des structures, des schémas de programme **stéréotypés,** qui ont pour vertu de tendre à réduire le nombre d'erreurs possibles aux seules erreurs « de détail », dont la correction ne remet pas en cause la conception ;
- que la stricte obéissance à de telles structures permet à l'inverse de **reformuler** un programme en **langue naturelle !** ce qui n'est pas toujours le cas quand le style d'un logiciel est par trop « artistique ».

Dans notre prochain article, nous serons conduits à compléter le tableau avec quelques structures additionnelles, très liées à l'organisation des données, qui sera notre thème principal.

A cette occasion, nous ferons connaissance plus intime avec le **Moniteur** et les joies de l'affichage (jeux, pendule digitale...), ainsi qu'avec l'important concept d'interruption. ■

**J.-M. COUR ***

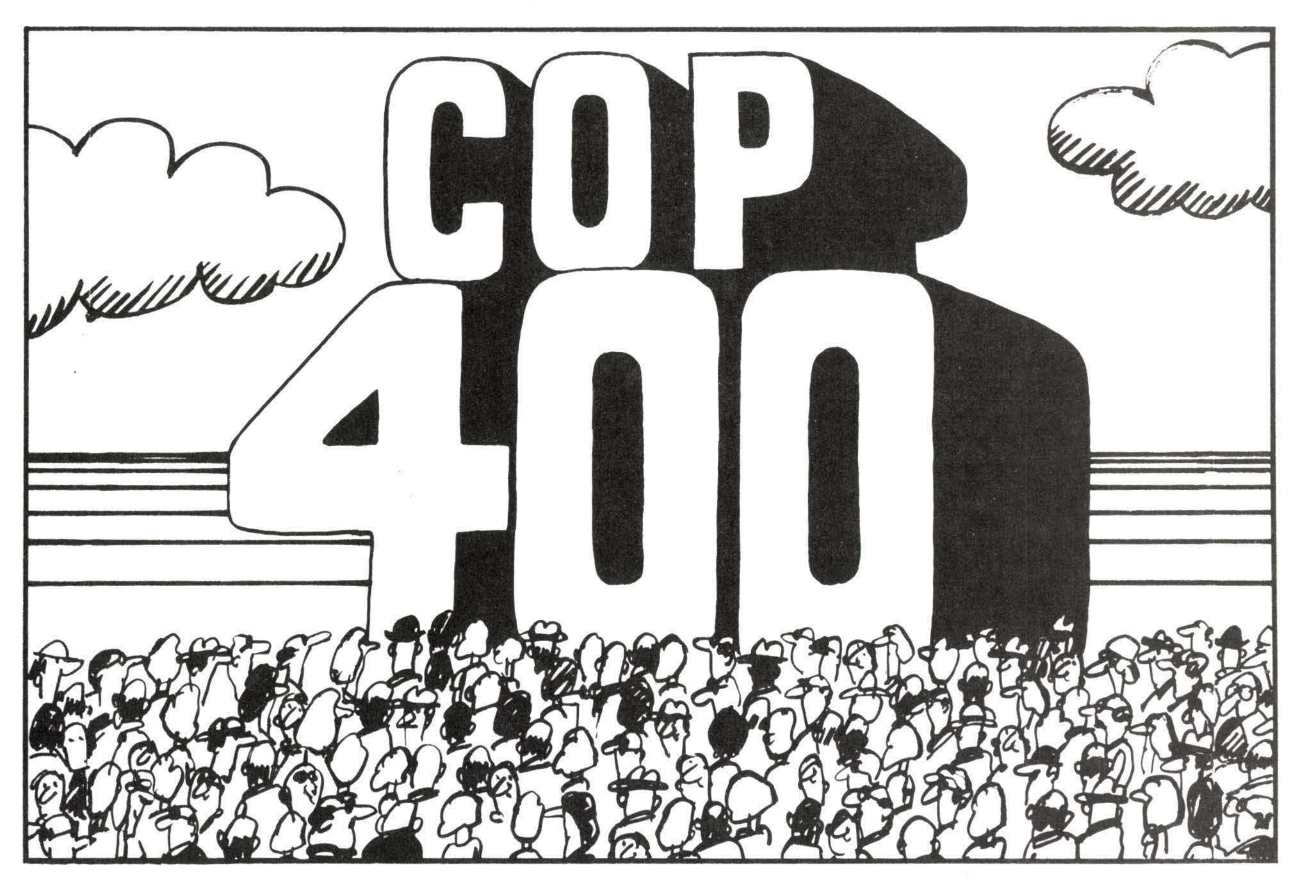
*Fig. 10*

**Programme :** EXEMPLE 4.3 **Auteur :** J.-M. COUR **Date :** 01/79 **Page :** 1/1

| Adresse | Traduction | Etiquette | Opération | Paramètres | Commentaires |
|---|---|---|---|---|---|
| 000 | 04 00 | DEBUT | LODI | R0, 0 | RESULTAT R0 ← 0 |
| 002 | B4 80 | | TPSU | SENSE | ATTENTE JUSQU'A |
| 004 | 9C 00 02 | | BCFA, 0 | DEBUT + 2 | SENSE = 1 |
| 007 | 07 ␣ | MESURE | LODI, R3 | X | ATTENTE PURE |
| 009 | 06 ␣ | | LODI, R2 | Y | |
| 00B | FB 7E | ATT | BDRR, R3 | ATT | |
| 00D | FA 7C | | BDRR, R2 | ATT | |
| 00F | 84 01 | | ADDI, R0 | 1 | MESURE ← MESURE + 1 |
| 011 | B4 80 | | TPSU | SENSE | JUSQU'A SENSE = 0 |
| 013 | 1C 00 07 | | BCTA, 0 | MESURE | |
| 016 | F0 | AFFICHER | WRTD | R0 | AFFICHAGE/VOYANTS |
| 017 | 40 | | HALT | | HALTE |

** Jean-Michel COUR anime la section « Micro-Informatique » dans la Société d'Ingénierie GIXI (Groupe CISI).*

# Une puissance nouvelle dans la technologie des micro-contrôleurs.




La série COP 400 : une nouvelle famille de micro-contrôleurs qui modifie radicalement les normes de l'industrie.

Tous les micro-contrôleurs de la série 400 possèdent un jeu d'instructions commun, le même système de développement et les mêmes fonctions de sortie. Tout ceci facilite évidemment la programmation. Vous pouvez donc maintenant utiliser le même langage et le même schéma pour toutes vos applications en réduisant au strict minimum la duplication du travail.

Et vous disposez de toute une famille de micro-contrôleurs parmi lesquels vous pourrez choisir ceux qui vous conviennent. Nous avons délibérément structuré nos microprocesseurs de manière à ce qu'ils s'adaptent à une vaste gamme de capacités de mémoire et de programmes. Et comme, tous nos micro-contrôleurs sont des dispositifs mono-circuits produits en grande série, nous sommes en mesure de les vendre à un prix vraiment minimum.

Grâce à toutes nos années d'expérience dans la fabrication des MOS, nous avons pu choisir la technologie qui convenait le mieux à la tâche en cause : canal "N" pour haute vitesse (COP 400), canal "N" faible puissance pour consommation réduite (COP 400 L) et CMOS (COP 400 C) pour consommation minimum.

## La technologie en pratique.

La série des micro-contrôleurs faible puissance COP 400 L est la seule à vous offrir :

| COP | 411 L | 410 L | 420 L | 421 L | 444 L |
|---|---|---|---|---|---|
| ROM 8 bits | 512 | 512 | 1024 | 1024 | 2048 |
| RAM 4 bits | 32 | 32 | 34 | 64 | 128 |
| Boîtier | 20 | 24 | 28 | 24 | 28 |

Nous offrons également des options entrée/sortie et d'horloge ainsi que d'autres circuits de la même famille, dans deux technologies supplémentaires.

Distribué dans toute la France par :

**GENERIM** – 91400 ORSAY - TÉL. : 907.78.78 • **R.T.F. DISTRONIQUE** – 92202 NEUILLY - TÉL. : 747.11.01.
**SCAIB** – 94150 RUNGIS - TÉL. : 687.23.13 • **APPLICATION ÉLECTRONIQUE** – 30001 NIMES - TÉL. : (16) 66.84.99.06.
**APPLICATION ÉLECTRONIQUE** – 31300 TOULOUSE - TÉL. : (16) 61.42.64.28 • **DEBELLE** – 38600 FONTAINE - TÉL. : (16) 76.26.56.54.
**STERC MAISSIAT** – 44010 NANTES - TÉL. : (16) 40.71.45.75 • **FIME** – 94262 FRESNES - TÉL. : 666.95.01.
**FACEN LILLE** – 59000 LILLE - TÉL. : (16) 20.96.93.07 • **FACEN NANCY** – 54140 HEILLECOURT - TÉL. : (16) 83.51.00.05.
**FACEN STRASBOURG** – 67450 MUNDOLSHEIM - TÉL. : (16) 88.20.20.80 • **FACEN ROUEN** – 76800 ST ÉTIENNE DU ROUVRAY - TÉL. : (16) 35.65.36.03 / 39.06.
**FACEN (CENTRE ADMINISTRATIF)** – 59290 WASQUEHAL - TÉL. : (16) 20.98.92.15.

**National Semiconductor**

28, rue de la Redoute - 92260 FONTENAY-AUX-ROSES - Tél. : 660.81.40

# NASCOM 1

*ou le portrait d'un sujet britannique au-dessus de tout soupçon*

COMME tout produit britannique le NASCOM 1, ordinateur personnel, possède des qualités typiquement anglaises. Il est racé, puissant, sophistiqué, étonnamment compétitif et raffiné.
Racé, parce qu'il descend directement d'une technologie éprouvée.
Puissant, parce qu'équipé du fameux micro-processeur Z 80.
Sophistiqué parce que ses auteurs l'ont conçu pour accepter, dès l'origine, toutes les extensions dont vous voudrez le doter ultérieurement : sortie vidéo, télétype, basic version mini ou étendue, RAM jusqu'à 64 K, floppy-disque, etc...
Compétitif, parce que comparé aux autres systèmes de sa classe, il reste très bon marché par les performances qu'il présente.
Raffiné, car comme tout bon anglais, il fait partie d'un club privé, dont, il vous ouvre toutes grandes les portes; vous y trouverez une bibliothèque de programmes déjà réalisés, mais rien ne vous empêchera d'y faire figurer les vôtres.
Ajoutez l'élégance, car vous pouvez l'habiller dans un rack d'aspect très professionnel, et, vous aurez mis la touche finale à ce portrait d'un sujet britannique au-dessus de tout soupçon.


**Le système de base, se raccordant directement à un téléviseur standard et à un magnéto cassette, comprend le clavier alphanumérique, et, la carte de base en kit, avec :**

- Micro-processeur Z 80
- Interface vidéo et TV
- Interface magnéto-cassette
- Inter face E/S série, boucle 20 MA ou RS 232
- Inter face parallèle 16 E/S
- RAM 1 K x 8 disponible utilisateur
- Moniteur NASBUG
- Connecteur d'extension

**Livré avec une documentation détaillée dont 2 manuels de montage et de logiciel en français ........................................ 2490 F/TTC**

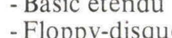
**Options disponibles en kit**

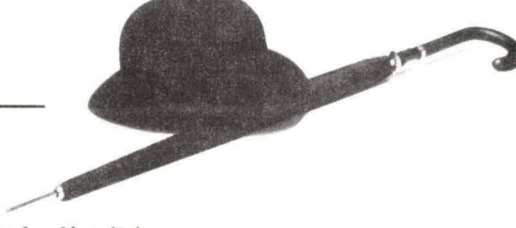
- Carte mémoire 8 K x 8
- Carte mémoire 16 K x 8
- Carte mémoire 32 K x 8
- Tiny Basic en EPROM
- Assembleur-éditeur
- Alimentation 2,2 A
- Carte Buffer-bus

**Options bientôt disponibles**

- Rack 19"
- Carte E/S supplémentaire
- Carte relais
- Alimentation 8 A


**Options en préparation**

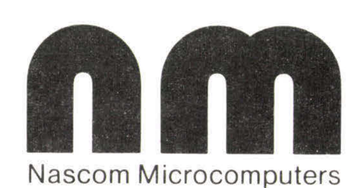

- Basic étendu
- Floppy-disque



**carte mémoire** 32 K x 8 **2980 F/TTC**

**Liste des dépositaires**

CSE 15, rue Clovis - 57000 METZ
DELOCK 4, rue Colbert - 59000 LILLE
ELECTROME 17, rue Fondaudège 33000 BORDEAUX
EQUIP. ELEC. EST 7, rue de la Loi 68100 MULHOUSE
FANATRONIC 35, rue de la Croix-Nivert - 75015 PARIS
FANATRONIC 2, bd du Sud-Est - 92000 NANTERRE
LISCO 43, Grand-Place - 38000 GRENOBLE
REBOUL 34, rue d'Arènes - 25000 BESANÇON
SELECTRONIC 14, Bd Carnot - 59000 LILLE
SELFCO 31, rue du Fossé des Treize - 67000 STRASBOURG
SOS T.V. AYZE - 74130 BONNEVILLE




**IMPORTATEUR**
**JCS COMPOSANTS**
**35, rue de la Croix Nivert 75015 PARIS - Tél. 306.93.69**

# Le 6800

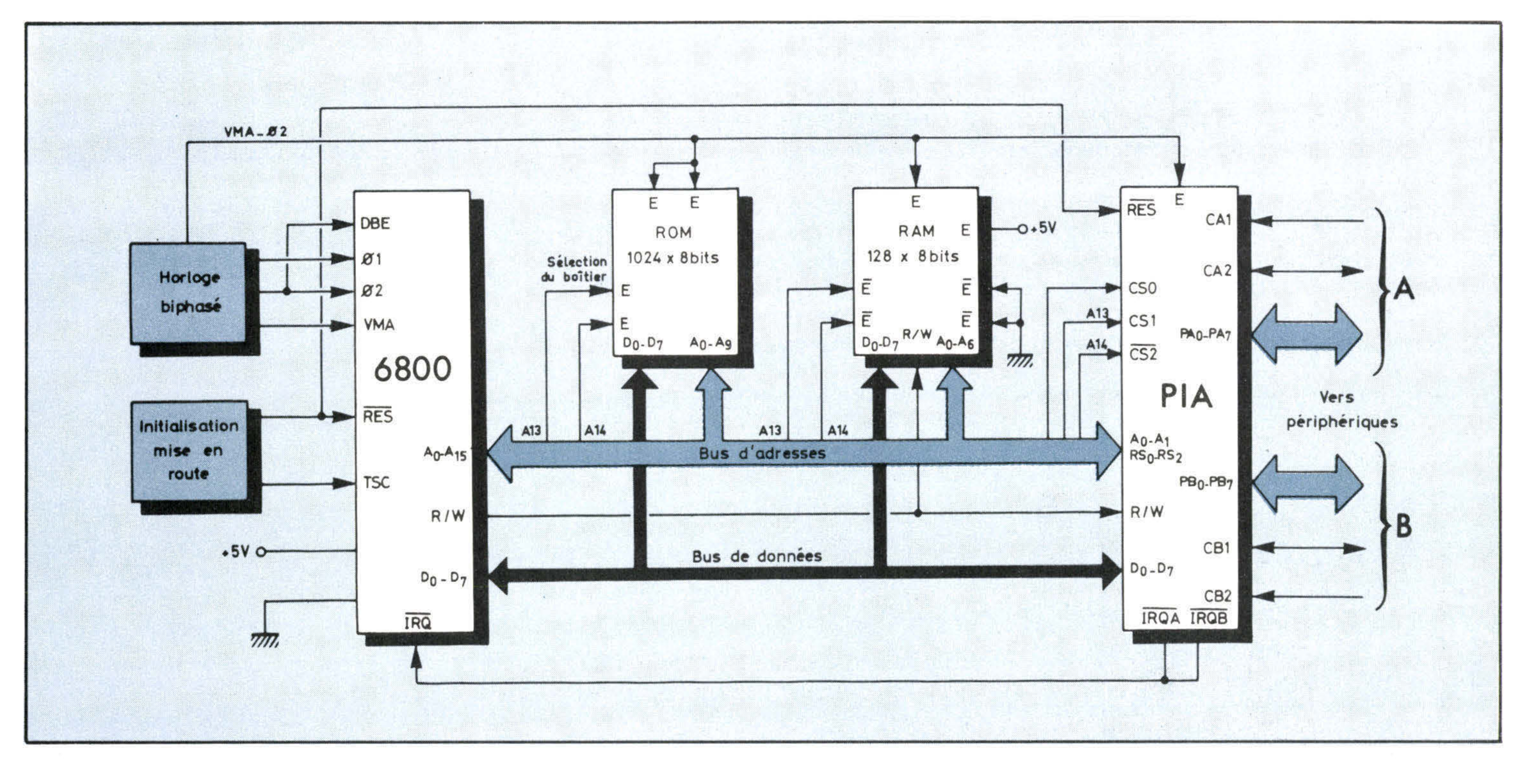



*Fig. 1. – Système minimum conçu autour des éléments de la famille 6800.*

Initialement étudié et commercialisé par Motorola, le 6800, l'un des microprocesseurs le plus connu et utilisé du marché, a été ensuite repris par des constructeurs comme Sescosem (réf. : SF.F 96800), A.M.I., Hitachi, qui constituent autant de secondes sources disponibles pour l'utilisateur.

Ce circuit intégré traite des mots de 8 bits et constitue une unité centrale complète en un seul boîtier DIL 40 broches. Comme la plupart des unités centrales actuellement commercialisées, le 6800 est réalisé en technologie MOS à canal N et porte au silicium.

En outre, de manière à faciliter la tâche du concepteur de systèmes, le constructeur a étudié toute une gamme de circuits directement compatibles avec le 6800.

Cette famille possède une tension d'alimentation unique de + 5 V ± 5 % qui la rend compatible TTL et qui permet un interfaçage aisé avec les bus. Elle se compose entre autre de circuits : d'adaptation aux périphériques, PIA 6820 (SF.F 96820) ; d'adaptation pour communications séries asynchrones, ACIA 6850 (SF.F 96850) ; d'adaptation pour communications séries synchrones SSDA, SF.F 96852 et de mémoires telles que la RAM de 128 octets 6810 (SF.F 96810) et la ROM 1024 octets 6830 (SF. F 96830).

Enfin, notons que dans sa version standard, le SF. F 96800 est piloté par une horloge de 1 MHz à deux phases séparées donc sans recouvrement. Dans le cas d'une application où des vitesses plus élevées sont requises, il convient d'employer le SF. F 96800 A de fréquence d'horloge 1,5 MHz ou le SF. F 96800 B dont la fréquence d'horloge typique est de 2 MHz.

## Caractéristiques essentielles du 6800

- Microprocesseur 8 bits.
- Tension d'alimentation unique : + 5 V.
- Bus de données bi-directionnel.
- Bus d'adresse de 16 bits - espace d'adressage de 64 K octets.
- Six registres internes.
- 72 instructions de longueur variable.
- Temps d'exécution de l'addition : 2 μs.
- Sept modes d'adressage : direct, relatif, immédiat, indexé, étendu, implicite et accumulateur.
- Pile externe de longueur variable.
- Vecteur d'interruption masquable.
- Interruption non masquable.
- Possibilités d'accès direct mémoire (DMA) et de configuration multiprocesseur.
- Possibilité d'arrêt et d'exécution pas à pas.

## Description des broches

| Réf. | Broches | Fonctions |
|---|---|---|
| $V_{ss}$ | 1 et 21 | 0 V |
| $V_{cc}$ | 8 | Alimentation : + 5 V ± 5 % |
| $\overline{\text{Halt}}$ | 2 | Arrêt du $\mu$P. |
| Ø 1 | 3 } | Signal d'horloge 2 phases |
| Ø 2 | 37 } | |
| $\overline{\text{IRQ}}$ | 4 | Demande d'interruption |
| VMA | 5 | Validation de l'adresse-mémoire. |
| $\overline{\text{NMI}}$ | 6 | Interruption non masquable |
| BA | 7 | Bus disponible |
| $A_0$-$A_{15}$ | 9-20<br>22-25 | Bus d'adresse |
| $D_0$-$D_7$ | 26-33 | Bus de donnée |
| R/W | 34 | Lecture/écriture |
| DBE | 36 | Activation du bus de donnée |
| TSC | 39 | Contrôle trois états |
| $\overline{\text{Reset}}$ | 40 | Remise à l'état initial |

*Fig. 3. - Schéma fonctionnel de l'unité centrale.*



## Caractéristiques générales

Livré dans un boîtier DIL à 40 broches **(fig. 2)**, le 6800 traite des

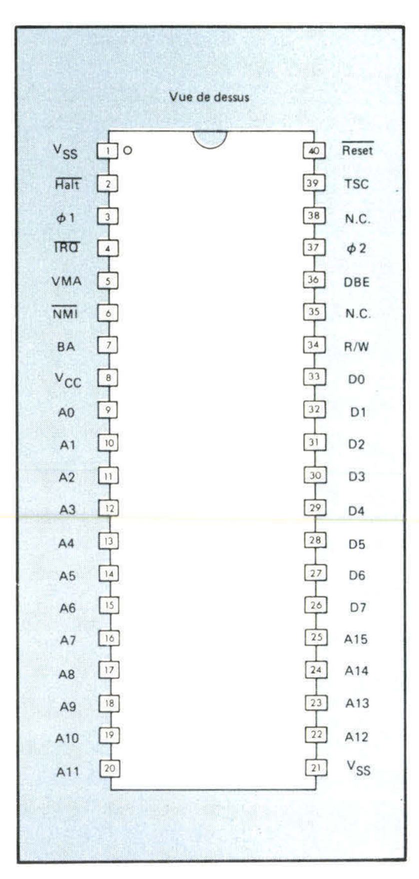



*Fig. 2. - Brochage du 6800.*

mots de 8 bits et réalise la fonction d'unité centrale pour la famille 6800 de Motorola ou SF.F 96800 de chez Sescosem.

Ainsi, la construction d'un système minimum n'exige que peu de circuits extérieurs. Une mémoire morte (ROM), une mémoire vive (RAM), une horloge et un circuit d'adaptation aux périphériques (PIA) permettent la construction du système représenté **figure 1.**

## Constitution

L'organisation générale du 6800 est donnée **figure 3.** Nous retrouvons ici, comme dans tous les microprocesseurs, l'unité arithmétique et logique (ALU) qui gère,

traite et manipule les données, l'unité de contrôle associée au registre d'instruction qui analyse et décode les instructions présentes sur le bus de données, la mémoire interne à l'unité centrale représentée par 6 registres mis à la disposition de l'utilisateur et les amplificateurs de sortie et de données aussi appelés buffers.

La **figure 4** montre les différents registres et les lignes d'entrées/sorties accessibles. Ces lignes d'entrées/sorties se composent des bus et des signaux suivants :

— Bus de données 8 bits bi-directionnel.
— Bus d'adresses 16 bits.
— Bus de contrôle représenté par R/W, VMA, $\overline{\text{IRQ}}$, $\varnothing_1$, $\varnothing_2$ et $\overline{\text{Reset}}$.
— Bus de commande : DBE, TSC, BA, $\overline{\text{Halt}}$, NMI et $\overline{\text{Reset}}$.

Le détail des 6 registres internes du 6800 est donné **figure 5.**

Les deux accumulateurs 8 bits référencés ACC A et ACC B sont employés pour mémoriser les opérandes (ou données) et les résultats de l'unité arithmétique et logique. Un registre d'index de 16 bits (IX) est utilisé comme index dans le mode d'adressage indexé et occasionnellement pour des transferts de données. Le pointeur de pile de 16 bits contient l'adresse de la position mémoire disponible dans une pile externe du type LIFO (dernier entré, premier sorti). Bien entendu le compteur de programme ou compteur ordinal est chargé de mémoriser l'adresse sur 16 bits de l'instruction à exécuter.

*Fig. 4. – Les registres internes et les lignes d'entrées/sorties du 6800.*

*Fig. 5. – Les registres programmables du microprocesseur. Le registre d'index, le compteur de programme et le pointeur de pile ont une longueur de 16 bits.*

Enfin, un registre d'état de 8 bits permet de disposer des 6 informations correspondantes.

Soit pour 5 d'entre elles, aux résultats d'une opération effectuée dans l'ALU :

- Bit 0 : C – retenue (du bit 7)
- Bit 1 : V – dépassement de capacité
- Bit 2 : Z – résultat nul
- Bit 3 : N – résultat négatif
- Bit 5 : H – demi-retenu (ou retenue du bit 3)

soit, pour le bit 4 de masquer les interruptions.

Les deux derniers bits, 6 et 7, inutilisés sont toujours au niveau logique « 1 ».

## Définition des signaux

La signification succincte de chacun des signaux d'entrée/sortie est donnée en encadré. Nous la reprenons ci-dessous de façon plus détaillée :

- 1 et 21 : $V_{ss}$ alimentation 0 V
- 8 : $V_{cc}$ alimentation + 5 V
- 2 : $\overline{\text{Halt}}$

Signal d'entrée

Lorsque $\overline{\text{Halt}}$ est au niveau haut (état marche), l'unité centrale exécute les instructions. Si $\overline{\text{Halt}}$ est au niveau bas (état arrêt) elle passe dans l'état arrêt ou de repos. $\overline{\text{Halt}}$ permet la commande de l'exécution du programme par une source externe. La sortie BA donne une indication sur l'état courant du microprocesseur. Si BA = 0, il est en mode exécution, si BA = 1 il est à l'arrêt.

- 3 et 37 : $\varnothing_1$ et $\varnothing_2$

Signal d'horloge 2 phases sans recouvrement

- 4 : $\overline{\text{IRQ}}$

Signal d'entrée

Demande d'interruption (Interrupt Request). Cette entrée demande l'exécution d'une séquence d'interruption quand $\overline{\text{IRQ}}$ est à « 0 ». Le microprocesseur achève l'exécution de l'instruction en cours avant de prendre en compte la demande d'interruption. Si, à cet instant, le Bit I du masque d'interruption du registre

d'état est à 0, le microprocesseur commencera une séquence d'interruption.

• 5 : VMA
Signal de sortie.

Validation de l'adresse mémoire (Valid Memory Adress). Lorsque VMA est à 1, une adresse valide est présente sur le bus d'adresse. Ce signal est utilisé pour sélectionner les circuits.

• 6 : NMI
Signal d'entrée.

Interruption non masquable (Non Masquable Interrupt). Le bit 1 du registre d'état est ici sans influence. Un front descendant appliqué à cette entrée, signale une interruption non masquable. Le microprocesseur achève l'exécution de l'instruction en cours avant de prendre en compte ce signal.

• 7 : BA
Signal de sortie.

Bus disponible (Bus Available). Au niveau logique « 1 », ce signal indique que le microprocesseur est à l'arrêt et que le bus d'adresse est disponible.

• 9 -20 : $A_0$-$A_{15}$ Bus d'adresse
• 22-25
Signaux de sortie.

Bus d'adresse 16 bits permettant d'adresser 64 K octets de mémoire. Dans l'état « haute impédance », ces sorties sont pratiquement à circuit ouvert.

• 26-33 : $D_0$-$D_7$
Bus de données.

Bus de données 8 bits bi-directionnel permettant le transfert de données entre microprocesseur et circuits mémoires ou périphériques - peuvent être placés à l'état « haute impédance ».

• 34 : R/W
Signal de sortie.

Le signal de lecture/écriture (Read/Write) indique aux circuits mémoires et périphériques que le microprocesseur est dans un état de lecture si R/W = 1 ou dans un état d'écriture lorsque R/W = 0. Au repos, ce signal est à l'état haut (lecture).

• 36 : DBE
Signal d'entrée.

Activation du bus de données (Data Bus Enable) ; signal de contrôle 3 états pour le bus de données. A l'état haut, elle active les amplificateurs de sortie du bus.

• 39 : TSC
Signal d'entrée.

Contrôle trois états (Three State Control), permet de faire passer dans l'état haute impédance les lignes d'adresses et la ligne de lecture/écriture (R/W).

Le bus de données n'est pas affecté par un TSC et possède sa propre ligne d'activation (DBE).

• 40 : $\overline{\text{Reset}}$
Signal d'entrée.

Remise à l'état initial. Après une mise sous tension ou une panne d'alimentation, cette entrée permet l'initialisation du microprocesseur.

Lorsque le $\overline{\text{Reset}}$ est à l'état haut, l'unité centrale commence sa séquence de démarrage en lisant les deux derniers octets de mémoire FFFE et FFFF. Les contenus de ces deux octets sont chargés dans le compteur de programme et correspondent à l'adresse du début du programme.

## Le logiciel

Le microprocesseur 6800 est doté d'un jeu de 72 instructions exécutables et assemblées, suivant le mode d'adressage choisi, en 1, 2 ou 3 octets.

Le premier octet suffit à identifier l'instruction et son mode d'adressage. Lorsqu'une instruction se compose de 2 ou 3 octets, le 2e octet ou les 2e et 3e octets contiennent un opérande (donnée), une adresse ou une information définissant une adresse.

Le jeu d'instruction contient des instructions d'arithmétique binaire et décimale, des instructions logiques, des instructions de décalage, de décalage circulaire, de chargement de registre (load), de rangement en mémoire (store), des instructions de branchement conditionnel ou inconditionnel, de traitement des interruptions et des instructions de manipulations de pile. Le **tableau 1** donne la liste alphabétique des instructions.

**Modes d'adressage**

Il existe 7 possibilités d'adressage pour chacune des instructions dont les modes sont les suivants :

• Adressage immédiat
• Adressage direct
• Adressage indexé
• Adressage étendu
• Adressage implicite
• Adressage relatif
• Adressage accumulateurs (ACCX).

Nous vous présentons ci-dessous une description succincte de chacun de ces modes d'adressage.

**Adressage immédiat :** (2 ou 3 octets)

Le premier octet de l'instruction contient le code opération et le deuxième octet l'opérande (dans les instructions LDS et LDX, l'opérande est contenue dans les 2e et 3e octets de l'instruction).

**Adressage direct :** (2 octets)

Ici, l'adresse de la donnée à rechercher est contenue dans le deuxième octet de l'instruction. Ainsi, ce mode d'adressage permet d'adresser les 256 premiers octets de la mémoire (de l'adresse 0 à 255).

**Adressage indexé :** (2 octets)

L'adresse contenue dans le 2e octet de l'instruction souvent nommé **déplacement**, est ajoutée à l'octet de poids faibles du registre d'index. Le résultat est utilisé pour adresser l'opérande à rechercher ou à stocker en mémoire.

**Adressage étendu :** (3 octets)

Cette instruction a une longueur de 3 octets. L'adresse en mémoire de la donnée est contenue dans le 2e octet (poids forts) et dans le 3e octet (poids faibles) de l'instruction.

**Adressage implicite :** (1 octet)

Peuvent être classées dans la catégorie « adressage implicite » les instructions :

• ne nécessitant pas d'adresse (ex : ABA : ajouter le contenu de l'accumulateur A à l'accumulateur B et placer le résultat dans A).
• nécessitant une adresse. L'adresse est ici contenue dans un registre interne au microprocesseur (ex : pointeur de pile).

Ainsi, aucune information n'est exigée pour fournir l'adresse.

**Adressage relatif :** (2 octets)

L'adresse contenue dans le 2e octet de l'instruction est ajoutée à l'octet de poids faible du compteur de programme plus deux. Les limites de l'adressage sont donc de - 126 à + 129 par rapport à l'instruction courante.

**Adressage d'accumulateur :** (1 octet)

La donnée est contenue soit dans l'accumulateur A soit dans l'accumulateur B.

**Le tableau 2** donne la liste des instructions relatives aux accumulateurs et à la mémoire. **Le tableau 3** indique les instructions spécifiques au registre d'état.

Chacun de ces tableaux présente les instructions, leurs mnémoniques, le code opération (OP) en hexadécimal en fonction du mode d'adressage, le nombre de cycles d'horloge nécessaires à l'exécution des instructions (~), le nombre d'octet de programme (#) et la signification symbolique des opérations effectuées.

Lorsqu'un bit du registre d'état est modifié après l'exécution d'une instruction, les symboles suivants permettent d'identifier son nouvel état.

R toujours mis à « 1 »
S toujours mis à « 0 »
↕ test : mis à 1 si vrai ; mis à zéro si non
• inchangé.

Les 6 bits présents dans le mot d'état sont référencés par des lettres :

H demi retenue (ou retenue du bit 3)
I masque d'interruption

| | | | |
|---|---|---|---|
| ABA | Addition de l'accumulateur B à l'accumulateur A | JSR | Saut à un sous-programme |
| ADC | Addition avec retenue | | |
| ADD | Addition | LDA | Chargement accumulateur |
| AND | « ET » logique | LDS | Chargement du pointeur de pile |
| ASL | Décalage arithmétique de un vers la gauche | LDX | Chargement du registre d'index |
| ASR | Décalage arithmétique de un vers la droite | LSR | Décalage logique vers la droite d'une position |
| BCC | Branchement s'il n'y a pas de retenue | NEG | Complément à deux (opposé) |
| BCS | Branchement s'il y a retenue | NOP | Passage en séquence (non opération) |
| BEQ | Branchement si égal (à zéro) | | |
| BGE | Branchement si supérieur ou égal à zéro | ORA | « OU » logique |
| BGT | Branchement si plus grand que zéro | | |
| BHI | Branchement si supérieur | PSH | Mise d'un octet dans la pile |
| BIT | Test de bits | PUL | Extraction d'un octet de la pile |
| BLE | Branchement si inférieur ou égal à zéro | | |
| BLS | Branchement si inférieur ou égal | ROL | Décalage circulaire à gauche |
| BLT | Branchement si inférieur à zéro | POR | Décalage circulaire à droite |
| BMI | Branchement si négatif | RTI | Retour de séquence d'interruption |
| BNE | Branchement si non nul | RTS | Retour de sous-programme |
| BPL | Branchement si positif ou nul | | |
| BRA | Branchement inconditionnel | SBA | Soustraction entre accumulateurs |
| BSR | Branchement à un sous-programme | SBC | Soustraction avec retenue |
| BVC | Branchement si pas de dépassement | SEC | Mise à un de la retenue |
| BVS | Branchement si dépassement | SEI | Mise à un du masque d'interruption |
| CBA | Comparaison des accumulateurs | SEV | Mise à un du bit de dépassement en complément à deux |
| CLC | Mise à zéro du bit de retenue | | |
| CLI | Mise à zéro du masque d'interruption | STA | Mise en mémoire d'un accumulateur |
| CLR | Mise à zéro | STS | Mise en mémoire du pointeur de pile |
| CLV | Mise à zéro du bit de dépassem. en compl. à deux | STX | Mise en mémoire du registre d'index |
| CMP | Comparaison | SUB | Soustraction |
| COM | Complément à un | SWI | Interruption programmée |
| CPX | Comparaison du registre d'index | | |
| | | TAB | Transfert de l'accumulateur A dans l'accumulateur B |
| DAA | Ajustement décimal sur l'accumulateur A | | |
| DEC | Décrémentation | TAP | Transfert de l'accumulateur A dans le Registre d'Etat |
| DES | Décrémentation du pointeur de pile | | |
| DEX | Décrémentation du registre d'index | TBA | Transfert de l'accumulateur A |
| | | TPA | Transfert du registre d'état dans l'accumulateur A |
| EOR | « OU » exclusif | TST | Test |
| | | TSX | Transfert du pointeur de pile dans le registre d'index |
| INC | Incrémentation | | |
| INS | Incrémentation du pointeur de pile | TXS | Transfert du registre d'index dans le pointeur de pile |
| INX | Incrémentation du registre d'index | | |
| JMP | Saut inconditionnel | WAI | Attente d'interruption |

*Tableau 1. - Liste alphabétique des 72 instructions du 6800.*

*Tableau 2. – Instructions relatives aux accumulateurs et à la mémoire.*

| INSTRUCTIONS | MNEMONIQUES | IMMEDIAT OP | ~ | # | DIRECT OP | ~ | # | INDEXE OP | ~ | # | ETENDU OP | ~ | # | IMPLICITE OP | ~ | # | OPERATION ARITHMETIQUE/ BOOLEENNE | 5 H | 4 I | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Addition | ADDA | 8B | 2 | 2 | 9B | 3 | 2 | AB | 5 | 2 | BB | 4 | 3 | | | | A + M → A | ↕ | • | ↕ | ↕ | ↕ | ↕ |
| | ADDB | CB | 2 | 2 | DB | 3 | 2 | EB | 5 | 2 | FB | 4 | 3 | | | | B + M → B | ↕ | • | ↕ | ↕ | ↕ | ↕ |
| Add. accumulateurs | ABA | | | | | | | | | | | | | 1B | 2 | 1 | A + B → A | ↕ | • | ↕ | ↕ | ↕ | ↕ |
| Add. avec retenue | ADCA | 89 | 2 | 2 | 99 | 3 | 2 | A9 | 5 | 2 | B9 | 4 | 3 | | | | A + M + C → A | ↕ | • | ↕ | ↕ | ↕ | ↕ |
| | ADCB | C9 | 2 | 2 | D9 | 3 | 2 | E9 | 5 | 2 | F9 | 4 | 3 | | | | B + M + C → B | ↕ | • | ↕ | ↕ | ↕ | ↕ |
| "ET" logique | ANDA | 84 | 2 | 2 | 94 | 3 | 2 | A4 | 5 | 2 | B4 | 4 | 3 | | | | A · M → A | • | • | ↕ | ↕ | R | • |
| | ANDB | C4 | 2 | 2 | D4 | 3 | 2 | E4 | 5 | 2 | F4 | 4 | 3 | | | | B · M → B | • | • | ↕ | ↕ | R | • |
| Test de bit | BITA | 85 | 2 | 2 | 95 | 3 | 2 | A5 | 5 | 2 | B5 | 4 | 3 | | | | A · M | • | • | ↕ | ↕ | R | • |
| | BITB | C5 | 2 | 2 | D5 | 3 | 2 | E5 | 5 | 2 | F5 | 4 | 3 | | | | B · M | • | • | ↕ | ↕ | R | • |
| Remise à zéro | CLR | | | | | | | 6F | 7 | 2 | 7F | 6 | 3 | | | | 00 → M | • | • | R | S | R | R |
| | CLRA | | | | | | | | | | | | | 4F | 2 | 1 | 00 → A | • | • | R | S | R | R |
| | CLRB | | | | | | | | | | | | | 5F | 2 | 1 | 00 → B | • | • | R | S | R | R |
| Comparaison | CMPA | 81 | 2 | 2 | 91 | 3 | 2 | A1 | 5 | 2 | B1 | 4 | 3 | | | | A – M | • | • | ↕ | ↕ | ↕ | ↕ |
| | CMPB | C1 | 2 | 2 | D1 | 3 | 2 | E1 | 5 | 2 | F1 | 4 | 3 | | | | B – M | • | • | ↕ | ↕ | ↕ | ↕ |
| Compar. accumulateurs | CBA | | | | | | | | | | | | | 11 | 2 | 1 | A – B | • | • | ↕ | ↕ | ↕ | ↕ |
| Complément à 1 | COM | | | | | | | 63 | 7 | 2 | 73 | 6 | 3 | | | | $\overline{M}$ → M | • | • | ↕ | ↕ | R | S |
| | COMA | | | | | | | | | | | | | 43 | 2 | 1 | $\overline{A}$ → A | • | • | ↕ | ↕ | R | S |
| | COMB | | | | | | | | | | | | | 53 | 2 | 1 | $\overline{B}$ → B | • | • | ↕ | ↕ | R | S |
| Complément à 2 (Négation) | NEG | | | | | | | 60 | 7 | 2 | 70 | 6 | 3 | | | | 00 – M → M | • | • | ↕ | ↕ | ① | ② |
| | NEGA | | | | | | | | | | | | | 40 | 2 | 1 | 00 – A → A | • | • | ↕ | ↕ | ① | ② |
| | NEGB | | | | | | | | | | | | | 50 | 2 | 1 | 00 – B → B | • | • | ↕ | ↕ | ① | ② |
| Ajustement décimal sur ACCA | DAA | | | | | | | | | | | | | 19 | 2 | 1 | Convertit le résult. de l'addit. de caract. BCD dans le format BCD | • | • | ↕ | ↕ | ↕ | ③ |
| Décrémentation | DEC | | | | | | | 6A | 7 | 2 | 7A | 6 | 3 | | | | M – 1 → M | • | • | ↕ | ↕ | 4 | • |
| | DECA | | | | | | | | | | | | | 4A | 2 | 1 | A – 1 → A | • | • | ↕ | ↕ | 4 | • |
| | DECB | | | | | | | | | | | | | 5A | 2 | 1 | B – 1 → B | • | • | ↕ | ↕ | 4 | • |
| "OU" exclusif | EORA | 88 | 2 | 2 | 98 | 3 | 2 | A8 | 5 | 2 | B8 | 4 | 3 | | | | A ⊕ M → A | • | • | ↕ | ↕ | R | • |
| | EORB | C8 | 2 | 2 | D8 | 3 | 2 | E8 | 5 | 2 | F8 | 4 | 3 | | | | B ⊕ M → B | • | • | ↕ | ↕ | R | • |
| Incrémentation | INC | | | | | | | 6C | 7 | 2 | 7C | 6 | 3 | | | | M + 1 → M | • | • | ↕ | ↕ | ⑤ | • |
| | INCA | | | | | | | | | | | | | 4C | 2 | 1 | A + 1 → A | • | • | ↕ | ↕ | ⑤ | • |
| | INCB | | | | | | | | | | | | | 5C | 2 | 1 | B + 1 → B | • | • | ↕ | ↕ | ⑤ | • |
| Chargem. accumulateurs | LDAA | 86 | 2 | 2 | 96 | 3 | 2 | A6 | 5 | 2 | B6 | 4 | 3 | | | | M → A | • | • | ↕ | ↕ | R | • |
| | LDAB | C6 | 2 | 2 | D6 | 3 | 2 | E6 | 5 | 2 | F6 | 4 | 3 | | | | M → B | • | • | ↕ | ↕ | R | • |
| "OU" logique | ORAA | 8A | 2 | 2 | 9A | 3 | 2 | AA | 5 | 2 | BA | 4 | 3 | | | | A + M → A | • | • | ↕ | ↕ | R | • |
| | ORAB | CA | 2 | 2 | DA | 3 | 2 | EA | 5 | 2 | FA | 4 | 3 | | | | B + M → B | • | • | ↕ | ↕ | R | • |
| Mise de données dans la pile | PSHA | | | | | | | | | | | | | 36 | 4 | 1 | A → $M_{SP}$, SP – 1 → SP | • | • | • | • | • | • |
| | PSHB | | | | | | | | | | | | | 37 | 4 | 1 | B → $M_{SP}$, SP – 1 → SP | • | • | • | • | • | • |
| Chargement d'accumulateurs à partir de la pile | PULA | | | | | | | | | | | | | 32 | 4 | 1 | SP + 1 → SP, $M_{SP}$ → A | • | • | • | • | • | • |
| | PULB | | | | | | | | | | | | | 33 | 4 | 1 | SP + 1 → SP, $M_{SP}$ → B | • | • | • | • | • | • |
| Décalage circulaire vers la gauche | ROL | | | | | | | 69 | 7 | 2 | 79 | 6 | 3 | | | | M | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ROLA | | | | | | | | | | | | | 49 | 2 | 1 | A (C ← b7 ← b0) | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ROLB | | | | | | | | | | | | | 59 | 2 | 1 | B | • | • | ↕ | ↕ | ⑥ | ↕ |
| Décalage circulaire vers la droite | ROR | | | | | | | 66 | 7 | 2 | 76 | 6 | 3 | | | | M | • | • | ↕ | ↕ | ⑥ | ↕ |
| | RORA | | | | | | | | | | | | | 46 | 2 | 1 | A (C → b7 → b0) | • | • | ↕ | ↕ | ⑥ | ↕ |
| | RORB | | | | | | | | | | | | | 56 | 2 | 1 | B | • | • | ↕ | ↕ | ⑥ | ↕ |
| Décalage arithmétique vers la gauche | ASL | | | | | | | 68 | 7 | 2 | 78 | 6 | 3 | | | | M | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ASLA | | | | | | | | | | | | | 48 | 2 | 1 | A (C ← b7 ← b0 ← 0) | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ASLB | | | | | | | | | | | | | 58 | 2 | 1 | B | • | • | ↕ | ↕ | ⑥ | ↕ |
| Décalage arithmétique vers la droite | ASR | | | | | | | 67 | 7 | 2 | 77 | 6 | 3 | | | | M | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ASRA | | | | | | | | | | | | | 47 | 2 | 1 | A (b7 → b0 → C) | • | • | ↕ | ↕ | ⑥ | ↕ |
| | ASRB | | | | | | | | | | | | | 57 | 2 | 1 | B | • | • | ↕ | ↕ | ⑥ | ↕ |
| Décalage logique vers la droite | LSR | | | | | | | 64 | 7 | 2 | 74 | 6 | 3 | | | | M | • | • | R | ↕ | ⑥ | ↕ |
| | LSRA | | | | | | | | | | | | | 44 | 2 | 1 | A (0 → b7 → b0 → C) | • | • | R | ↕ | ⑥ | ↕ |
| | LSRB | | | | | | | | | | | | | 54 | 2 | 1 | B | • | • | R | ↕ | ⑥ | ↕ |
| Mise en mémoire de l'accumulateur | STAA | | | | 97 | 4 | 2 | A7 | 6 | 2 | B7 | 5 | 3 | | | | A → M | • | • | ↕ | ↕ | R | • |
| | STAB | | | | D7 | 4 | 2 | E7 | 6 | 2 | F7 | 5 | 3 | | | | B → M | • | • | ↕ | ↕ | R | • |
| Soustraction | SUBA | 80 | 2 | 2 | 90 | 3 | 2 | A0 | 5 | 2 | B0 | 4 | 3 | | | | A – M → A | • | • | ↕ | ↕ | ↕ | ↕ |
| | SUBB | C0 | 2 | 2 | D0 | 3 | 2 | E0 | 5 | 2 | F0 | 4 | 3 | | | | B – M → B | • | • | ↕ | ↕ | ↕ | ↕ |
| Soustract. accumulat. | SBA | | | | | | | | | | | | | 10 | 2 | 1 | A – B → A | • | • | ↕ | ↕ | ↕ | ↕ |
| Soustract. avec retenue | SBCA | 82 | 2 | 2 | 92 | 3 | 2 | A2 | 5 | 2 | B2 | 4 | 3 | | | | A – M – C → A | • | • | ↕ | ↕ | ↕ | ↕ |
| | SBCB | C2 | 2 | 2 | D2 | 3 | 2 | E2 | 5 | 2 | F2 | 4 | 3 | | | | B – M – C → B | • | • | ↕ | ↕ | ↕ | ↕ |
| Transfert entre accumul. | TAB | | | | | | | | | | | | | 16 | 2 | 1 | A → B | • | • | ↕ | ↕ | R | • |
| | TBA | | | | | | | | | | | | | 17 | 2 | 1 | B → A | • | • | ↕ | ↕ | R | • |
| Test (zéro ou négatif) | TST | | | | | | | 6D | 7 | 2 | 7D | 6 | 3 | | | | M – 00 | • | • | ↕ | ↕ | R | R |
| | TSTA | | | | | | | | | | | | | 4D | 2 | 1 | A – 00 | • | • | ↕ | ↕ | R | R |
| | TSTB | | | | | | | | | | | | | 5D | 2 | 1 | B – 00 | • | • | ↕ | ↕ | R | R |
| | | | | | | | | | | | | | | | | | | H | I | N | Z | V | C |

Remarque : Les instructions utilisant le mode d'adressage d'accumulateur sont incluses dans la colonne pour l'adressage implicite.

*Tableau 3. – Instructions spécifiques du registre d'état.*

| INSTRUCTIONS | MNEMONIQUE | IMPLICITE OP | ~ | # | OPERATION BOOLEENNE | 5 H | 4 I | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Mise à zéro du bit de retenue | CLC | 0C | 2 | 1 | 0 → C | • | • | • | • | • | R |
| Mise à zéro du masque d'interruption | CLI | 0E | 2 | 1 | 0 → I | • | R | • | • | • | • |
| Mise à zéro du bit de dépassement en complément à deux | CLV | 0A | 2 | 1 | 0 → V | • | • | • | • | R | • |
| Mise à un de la retenue | SEC | 0D | 2 | 1 | 1 → C | • | • | • | • | • | S |
| Mise à un du masque d'interruption | SEI | 0F | 2 | 1 | 1 → I | • | S | • | • | • | • |
| Mise à un du bit de dépassement en complément à deux | SEV | 0B | 2 | 1 | 1 → V | • | • | • | • | S | • |
| Transfert de l'accumulateur A dans le Registre d'Etat | TAP | 06 | 2 | 1 | A → CCR | 12 | | | | | |
| Transfert du Registre d'Etat dans l'accumulateur A | TPA | 07 | 2 | 1 | CCR → A | • | • | • | • | • | • |

N négatif (bit de signe)

Z zéro

V dépassement (en complément à 2)

C retenue du bit 7.

Notez que, dans ces tableaux, les transferts sont indiqués par une flèche (→), que $M_{sp}$ est le contenu de l'octet mémoire adressé par le pointeur de pile, que SP est le pointeur de pile, qu'un 0 est un bit à 0 et que 00 représente un octet à zéro. ■

# Programme générateur de phrases aléatoires

A partir de 5, 10, etc., phrases de départ, ce programme est capable, par un mécanisme de mélange aléatoire, de générer 100, 1 000 phrases ou plus.

Il peut avoir deux utilités à son actif :

- D'une part, il permet de se distraire. Nous n'avons pas poussé le vice jusqu'à donner à l'ordinateur des phrases de départ plus ou moins « pimentées », mais vous pouvez le faire chez vous. Le défoulement est garanti !
- Aussi curieux que cela puisse paraître, cette méthode de création ludique * de proposition rejoint les jeux d'inventions. En effet, si l'on donne au départ une phrase du type « la radio est à transistors » et que l'on reçoit à un certain moment un « La radio aime les diodes tunnel », ceci peut parfaitement constituer le point de départ d'une découverte.

Analysons donc de plus près ce programme à jouer ou à inventer :

Nous avons opté pour les propositions simples, du type sujet, verbe, complément. Bien entendu, il est aisé de réaliser des chaînages, ou des phrases à un ou plusieurs compléments, sujets, etc.

A la ligne 70, on prévoit trois tableaux de caractères, S $, comme sujet, P $, comme prédicat * et C $ comme complément. Nous réservons 10 places pour autant de mots de chaque type. Ce nombre de places peut d'ailleurs être différent suivant le désir de l'utilisateur. Attention néanmoins à la taille-mémoire, car une vingtaine d'octets par mot seront automatiquement réservés quelle que soit votre machine (il s'agit de caractères).

Pour des tableaux de 10 éléments nous obtenons 1 000 combinaisons possibles ce qui vous permettra de lire les phrases composées par votre ordinateur pendant plus d'une heure.

Par la suite, ligne 80, 90, 100, une boucle de lecture permet de remplir les trois tableaux de caractères avec des propositions de départ situées en DATA, vers la fin du programme, lignes 250 à 290, etc. Très faible sera la probabilité de retrouver les propositions telles qu'elles figurent initialement en DATA. Nous pouvons en conséquence y inscrire des mots tout simplement.

Une grande boucle, ligne 105, sert à fabriquer des phrases. Pour choisir un mot de façon aléatoire, on utilise un sous-programme situé en 1 000 qui tire au hasard un nombre parmi les indices possibles d'un tableau S $, P $, ou C $, selon la formule *.

N = INT (« Taille Max Tableau » x RND (0) + 1)

Au retour de ce sous-programme la valeur de l'indice indiquera le mot à imprimer, qu'il s'agisse du tableau de sujets S $, prédicats ou compléments (lignes 190, 210 et 230).

Tel est donc le contenu de la boucle d'impression.

Remarquons cependant trois détails significatifs : deux «;» à la fin de l'ordre d'impression du sujet et du prédicat, lignes 190 et 210 et l'absence de point virgule après l'impression du complément, ligne 230. Ceci signifie l'impression successive de trois chaînes de caractères qui seront contenues en DATA et le passage à la ligne suivante après l'impression du troisième. En conséquence, nous devons prévoir un blanc après chaque mot en DATA et même en début de mot pour « aérer » un peu la phrase conçue par le micro-ordinateur.

Place maintenant aux exemples .

Essayez ce programme avec différents mots de votre invention ; nous l'avons testé sur notre micro-ordinateur « Micro-Systèmes 1 ». Faites-nous part des phrases les plus amusantes composées par votre machine. Nous serions heureux de publier les meilleures d'entre elles. ■

** Ludique : qui est relatif au jeu.*
** Prédicat : attribut d'un mot, d'une proposition.*
** INT : integer est la valeur entière de l'expression entre parenthèse, après calcul.*
*RND : Random, fonction donnant un nombre aléatoire compris entre 0 et 1.*
*Pour tirer au sort un nombre de 1 à X par exemple, on met la taille maximum du tableau égale à X et N prendra toutes les variables de 1 à X inclus.*

```
0010 REM ...CE PROGRAMME FABRIQUE DES PHRASES DU TYPE
0020 REM ...SUJET(I)+PREDICAT(J)+COMPLEMENT(K) , I,J,K
0030 REM ...ETANT CHOISIS AU HASARD
0040 REM ... LA RECETTE EST SIMPLE :  ON PREND UNE
0050 REM ... POIGNEE DE SUJETS, VERBES ET COMPLEMENTS
0060 REM ... ET ON MELANGE.
0070 DIM S$(10), P$(10), C$(10)
0080 FOR I=1 TO 5
0090 READ S$(I), P$(I), C$(I)
0100 NEXT I
0105 FOR J=1 TO 100
0110 REM ... "J" EST LE NOMBRE TOTAL DE PROPOSITIONS A
0120 REM ...FABRIQUER. A CE STADE LE PROGRAMME REMPLIT
0130 REM ...TROIS TABLEAUX DE 5 ELEMENTS : DES SUJETS (S$)
0140 REM ... PREDICATS (P$) ET COMPLEMENTS (C$).
0150 REM ... LE NOMBRE TOTAL DE COMBINAISONS EST , BIEN
0160 REM ...ENTENDU DE 5*5*5=125. NOUS N'AFFICHERONS QU'UNE
0170 REM ...PARTIE........
0180 GOSUB 1000
0190 PRINT S$(N);
0200 GOSUB 1000
0210 PRINT P$(N);
0220 GOSUB 1000
0230 PRINT C$(N)
0240 NEXT J
0250 DATA LE THON , C'EST , BON
0260 DATA CYRIL , DANSE , COMME UN GRAND
0270 DATA PETITE MAMAN , EMBRASSE , LE PERE NOEL
0280 DATA LE PORT , S'ENDORT , SOUS LA BRUINE
0290 DATA LE TOIT , ME CACHE , LE SOLEIL
0300 END
1000 REM ...SOUS-PROGRAMME DE TIRAGE AU SORT
1010  N=INT(5*RND(0)+1)
1020 RETURN

READY
#RUN
PETITE MAMAN DANSE LE PERE NOEL
CYRIL S'ENDORT LE PERE NOEL
LE PORT C'EST LE PERE NOEL
LE PORT EMBRASSE SOUS LA BRUINE
PETITE MAMAN DANSE COMME UN GRAND
LE THON S'ENDORT BON
LE TOIT ME CACHE LE SOLEIL
LE THON DANSE LE PERE NOEL
LE PORT DANSE LE SOLEIL
LE PORT ME CACHE COMME UN GRAND
CYRIL C'EST SOUS LA BRUINE
PETITE MAMAN DANSE COMME UN GRAND
PETITE MAMAN EMBRASSE LE PERE NOEL
LE THON C'EST BON
PETITE MAMAN C'EST BON
PETITE MAMAN S'ENDORT BON
CYRIL C'EST LE SOLEIL
PETITE MAMAN DANSE BON
LE THON EMBRASSE COMME IN GRAND
LE PORT EMBRASSE LE SOLEIL
LE THON ME CACHE BON
LE THON EMBRASSE LE SOLEIL
....
```



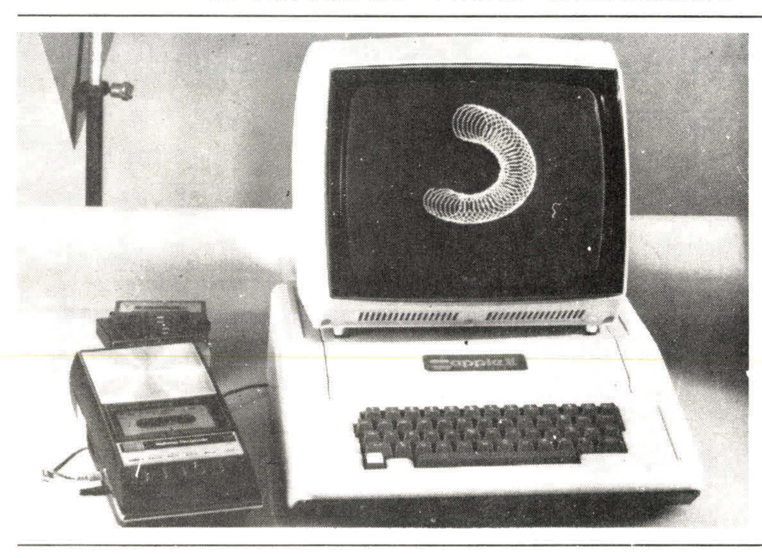

**DISTRIBUTEURS RECHERCHÉS DANS TOUTE LA FRANCE**

# MICRO-ORDINATEUR COMPUCOLOR



- MICROPROCESSEUR 8080 A.
- MEMOIRE RAM DE 8 A 32 Ko.
- BASIC 16 K RESIDENT
- INTERFACE RS 232 C.
- ECRAN DE VISUALISATION (64 c x 16/32).
- 8 COULEURS.
- TRACE GRAPHIQUE.
- 1 UNITE DE MINI-DISQUETTE.
- 64 CARACTERES SPECIAUX.
- OPTION IMPRIMANTE.

PRIX DE VENTE : 11.800 F HT.
COMPRENANT : ECRAN DE VISUALISATION AVEC 8 COULEURS – 1 UNITE DE MINI-DISQUETTE INTEGREE – CLAVIER COMPLET – 8 K RAM – BASIC RESIDENT.

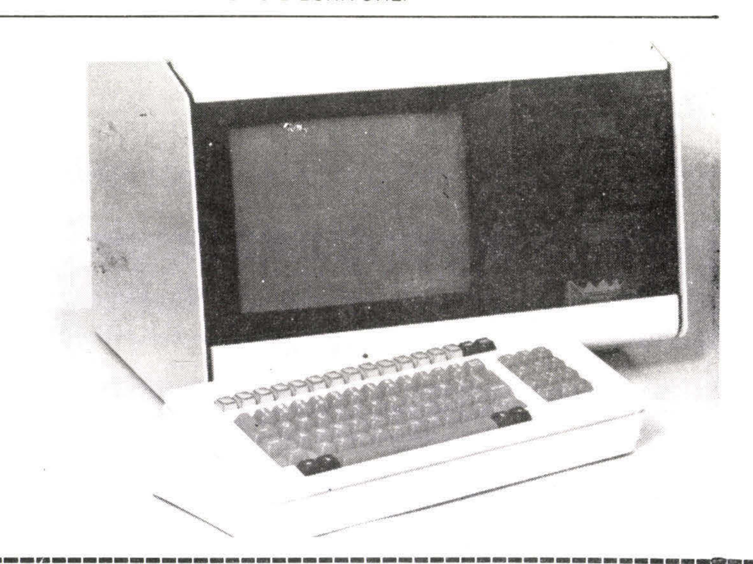


# MICRO-ORDINATEUR APPLE-II

- MICROPROCESSEUR ROCKWELL 6502 - RAM EXTENSIBLE DE 4 K A 48 K.
- BASIC - MONITEUR - ASSEMBLEUR - DESASSEMBLEUR (ROM).
- SORTIE VIDEO 24 LIGNES/40 COLONNES.
- GRAPHIQUES FINS EN COULEURS SUR T.V. (RVB-SECAM).
- INTERFACES MAGNETOPHONE ET ENTREES ANALOGIQUES - HAUT-PARLEUR INCORPORE.
- 8 PERIPHERIQUES CONNECTABLES DONT :
  - IMPRIMANTE, MODEM, CARTE DE COMMUNICATION RS 232
  - CARTE DE RECONNAISSANCE VOCALE (32 MOTS QUELCONQUES)
  - FLOPPY DISQUES (1 A 14 FOIS 116 Ko).

* DOS : FICHIERS DE DONNÉES EN ACCÈS SÉQUENTIEL INDEXÉ PROGRAMMATHIQUE/CHAINAGE DES PROGRAMMES/PROTECTIONS D'ÉCRITURE.

# MICRO-ORDINATEUR I.S.T.C. 5000



- MICROPROCESSEUR Z80 - RAM de 32 K à 64 K.
- SORTIE VIDEO 24 LIGNES/80 COLONNES.
- GENERATEUR DE CARACTERES PROGRAMMABLE.
- 1 OU 2 MINI-FLOPPY DISQUES (DOUBLE FACE) INTEGRES.
- DOS-EDITEUR DE TEXTE.
- MACRO ASSEMBLEUR.
- BASIC ETENDU (IF THEN ELSE, WHILE, PRINTUSING).
- FORTRAN IV ANSI
- EDITEUR DE LIENS POUR MODULES FORTRAN.
- 2 A 5 CONNECTEURS BUS S-100.
- INTERRUPTIONS CHAINEES AVEC PRIORITES (8 NIVEAUX).
- CARTE DE COMMUNICATION (SYNCHRONE/ASYNCHRONE).
- IMPRIMANTE AVEC INTERFACE.

---

Micro-Systèmes

**Bon réponse à retourner à :** I.S.T.C., 7 à 11, rue Paul-Barruel, 75015 Paris. Tél. : 306.46.06.

Raison Sociale ........................................

Activité ........................................

Nom et Fonction ........................................

........................................

Intéressé par : COMPUCOLOR ☐
APPLE II ☐ I.S.T.C. 5000 ☐

Adresse ........................................

........................................

Tél ........................................

☐ UNE DEMONSTRATION ☐ LA VISITE D'UN COMMERCIAL

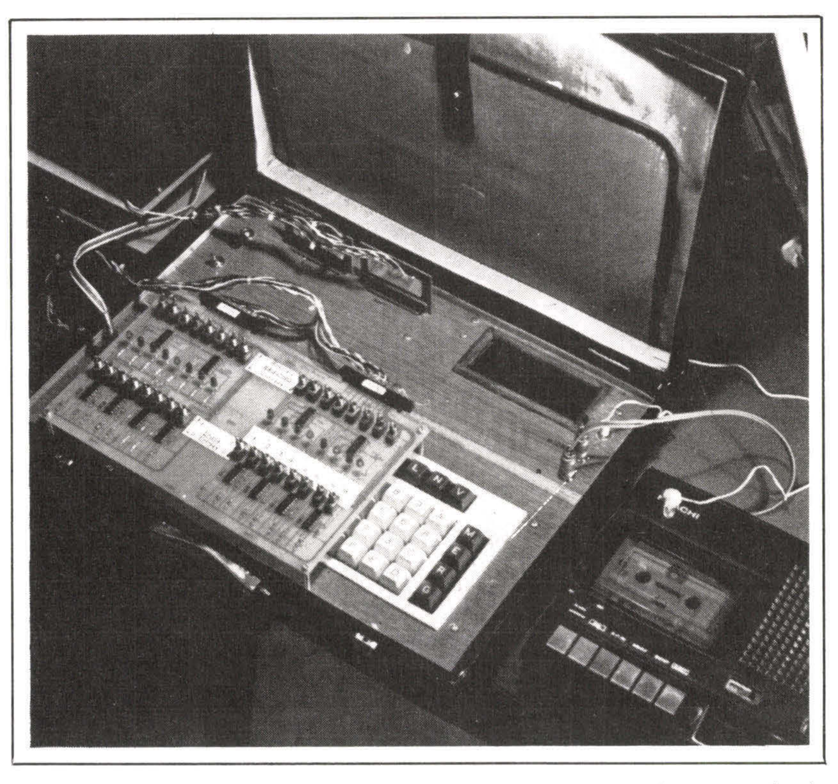
# Programme de jeu du « Master Mind »

**Le programme de jeu du « Master Mind » utilise le kit MK 2 de Motorola ou tout autre système construit autour d'un 6800 et doté d'au moins 512 octets de mémoire.**

**Ce jeu consiste à trouver une combinaison de quatre couleurs prises parmi six et proposées par l'un des joueurs appelé « codeur ». Une couleur peut être présente plusieurs fois dans la combinaison. Le « décodeur » propose au codeur une combinaison de son choix et ce dernier lui indique le nombre de couleurs effectivement présentes dans la combinaison initiale. Il indiquera séparément le nombre de couleurs correctement placées et celui de celles qui ne le sont pas.**

**Si, par exemple le codeur a choisi la combinaison bleu, bleu, rouge, vert, lorsque le décodeur annonce rouge, vert, noir, vert, la réponse du codeur sera d'une couleur correcte bien placée (le vert) et une couleur correcte mal placée (le rouge).**

**On compte le nombre d'essais nécessaires au décodeur pour trouver la combinaison proposée par le codeur et on permute les rôles. Le gagnant est celui qui totalise le moins d'essais.**



*Vue générale du micro-ordinateur placé dans un attache-case, avec son alimentation séparée à gauche et le magnétophone à cassette à droite. On distingue l'écouteur permettant d'écouter les commentaires entre les programmes enregistrés. La carte d'entrées/sorties du micro-ordinateur comporte 16 interrupteurs et 16 LEDS correspndant aux portes A et B (8004 et 8006) du PIA. Cette carte universelle, réalisée par notre auteur, permet de simuler un problème destiné à contrôler le programme en étude (ici le Master Mind). Un interrupteur sur chaque ligne, programme celle-ci en entrée (interrupteur) ou en sortie (LED). Dans le cas du Master Mind, on utilise le port B (8006) et les interrupteurs, de gauche à droite, sont 1, 2, 3, 4, 5, N, M.*

## Jeu de « Master Mind » avec un micro-ordinateur

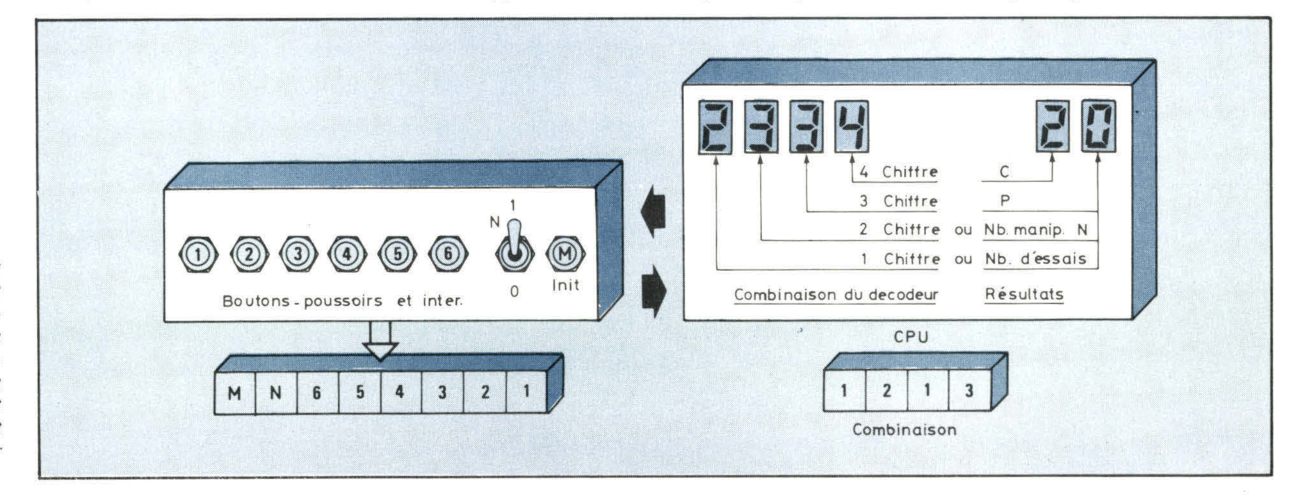
Le micro-ordinateur joue le rôle du codeur et propose une combinaison de quatre chiffres pris parmi les suivants : 1, 2, 3, 4, 5, 6.

Le décodeur affiche une combinaison de son choix et le micro-ordinateur, après comparaison avec sa propre combinaison, lui indique le nombre de chiffres corrects mais mal placés et celui des bien placés. Il compte également le nombre d'essais du décodeur.

## Organisation matérielle

Le programme qui va être étudié, pourra être utilisé par tout possesseur d'un kit Motorola MK 2 ou équivalent, auquel on ajoutera une planche de boutons-poussoirs.

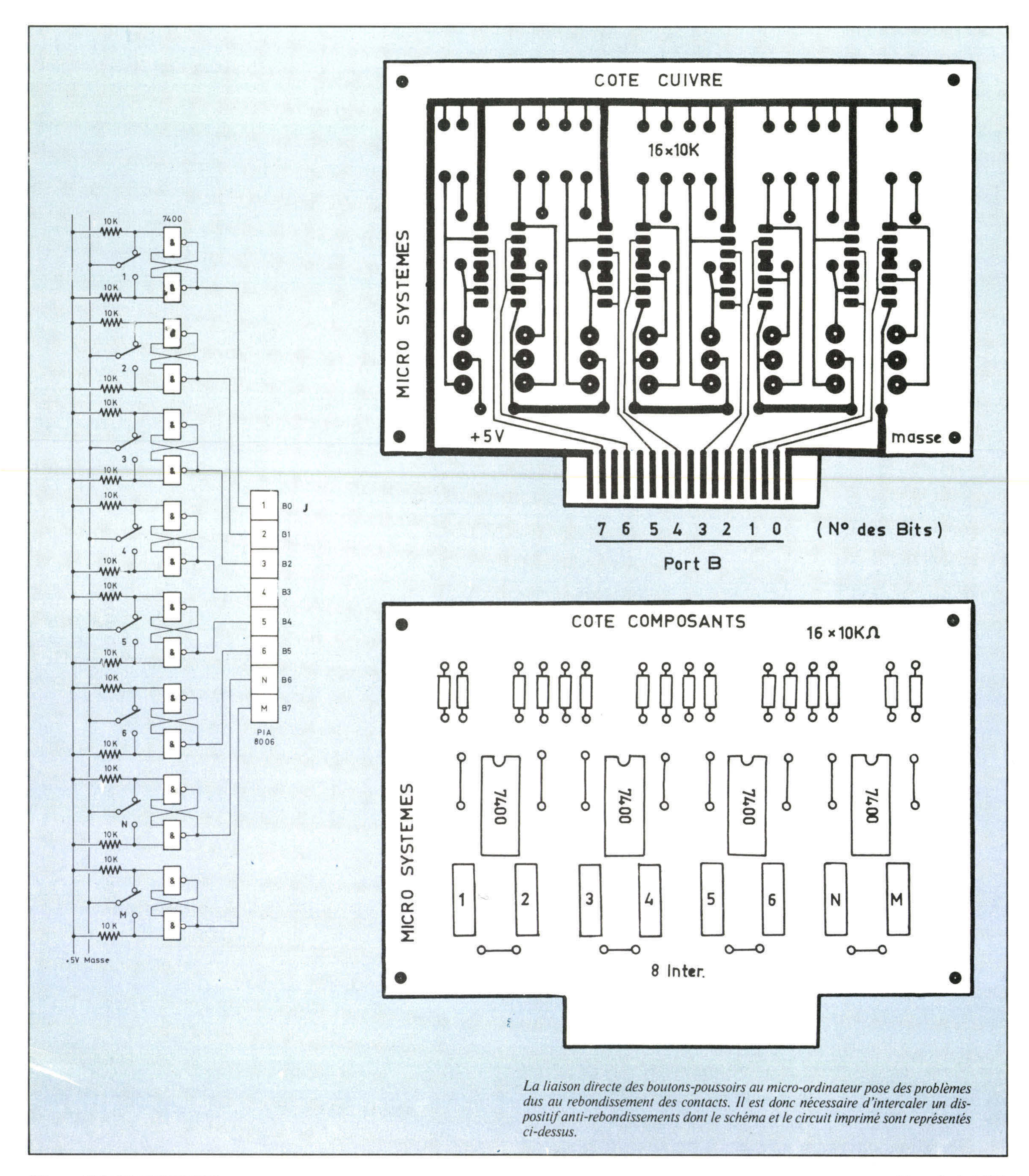
Le programme étant du type conversationnel, ces BP permettront la communication entre le décodeur et le micro-calculateur qui répondra par l'intermédiaire de son affichage **(fig. 1).**




*Les boutons poussoirs et l'affichage permettent le dialogue homme-machine. Le port B du PIA « utilisateur » (broches 7 à 14 du connecteur $J_1$) dont l'adresse est 8006 est utilisé pour assurer cette communication.*

Dans le jeu de « Master Mind », le micro-ordinateur propose une combinaison de quatre chiffres que vous devez découvrir.






*La liaison directe des boutons-poussoirs au micro-ordinateur pose des problèmes dus au rebondissement des contacts. Il est donc nécessaire d'intercaler un dispositif anti-rebondissements dont le schéma et le circuit imprimé sont représentés ci-dessus.*

Le clavier ne sera utilisé que pour lancer le programme initialement.

## Fonctionnement

● Après lancement du programme par affichage de l'adresse de début (000A) et appui fugitif sur la touche G du clavier du kit, les afficheurs font apparaître les chiffres 0000 00.

● Une action fugitive sur le bouton poussoir M initialise le jeu.

● Quatre manipulations de l'interrupteur N inscrivent dans la mémoire, la combinaison « aléatoire » que le décodeur devra découvrir. Le nombre de manipulations est indiqué sur l'afficheur de droite.

● Une cinquième manipulation de N prépare le micro-ordinateur à recevoir la première combinaison du décodeur.

● Celui-ci appuie sur les touches de 1 à 6 suivant son goût, sa combinaison s'inscrit au fur et à mesure sur les quatre premiers afficheurs.

● Le basculement de N en position 1 entraîne la comparaison de la combinaison du décodeur avec celle du calculateur. Le résultat de cette comparaison s'inscrit sur les deux afficheurs de droite.

● Le retour de N en position 0, prépare le calculateur pour la réception d'une nouvelle combinaison du décodeur et le cycle recommence.

● Quand la combinaison du décodeur est conforme à celle du calculateur, le chiffre 4 (quatre chiffres corrects et bien placés) et le nombre d'essais du décodeur pour arriver à la solution, s'inscrivent alternativement sur le dernier afficheur, à droite.

● Le rappel de N en position 0, puis, un appui fugitif sur le BP M, réinitialise le système qui est prêt pour une nouvelle partie.

## Conception générale du programme

Il faut comparer chaque chiffre de la combinaison du décodeur avec tous les chiffres de la combinaison du calculateur et déterminer ainsi :

● Si les deux chiffres comparés sont égaux.

● Si ces deux chiffres occupent le même rang.

● Il faut éviter d'effectuer une deuxième comparaison pour un des chiffres de la combinaison du décodeur quand ce chiffre a été trouvé égal à l'un de ceux de la combinaison du calculateur.

Pour cela, on associera un drapeau à chacun des chiffres des deux combinaisons et la comparaison ne sera effectuée que si les deux drapeaux sont à l'état 0. Dans le cas contraire, si l'un des drapeaux est à l'état 1, la comparaison sera interdite, et on passera au chiffre suivant.

### Organisation générale des variables

On désigne par $A_i$ les chiffres du décodeur et par $B_j$ les chiffres du calculateur.

D'autre part, $d_i$ sera le drapeau affecté à $A_i$ et $D_j$ celui affecté à $B_j$. C sera le nombre de chiffres corrects mais mal placés et P celui des chiffres corrects et bien placés. Les nombres i et j sont compris entre 1 et 4 **(fig. 2).**

### Exemple numérique illustrant ces opérations

Comme le montre la **figure 3 a,** on compare d'abord les chiffres de même rang et on positionne les drapeaux, le cas échéant.

Puis on compare chaque chiffre du décodeur à chacun des chiffres du calculateur en tenant compte de l'état des drapeaux correspondants **(fig. 3 b).**

Le résultat de la comparaison est donc de deux chiffres corrects, mal placés et un chiffre correct, bien placé.

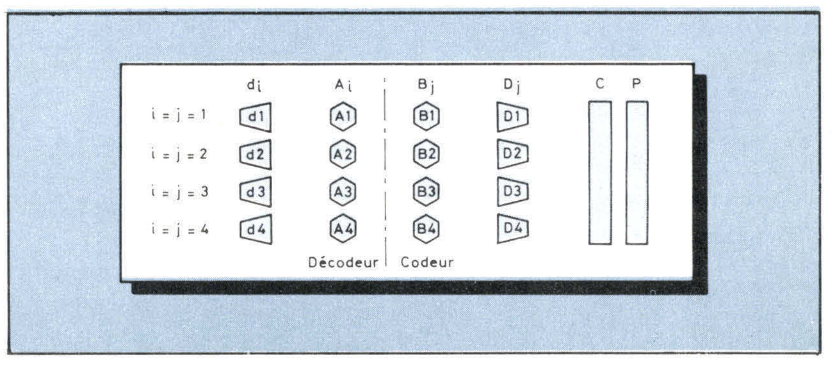



*Fig. 2. – Organisation générale des variables : les Ai représentent les chiffres du décodeur (utilisateur) et les Bj ceux de micro-ordinateur. C indique le nombre de chiffres corrects mais mal placés et P le nombre de chiffres corrects et bien placés.*

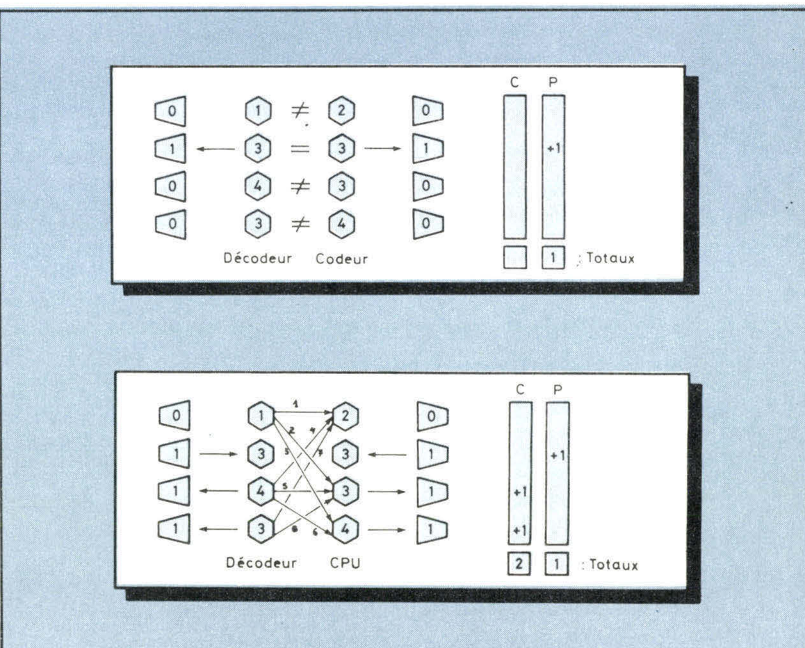
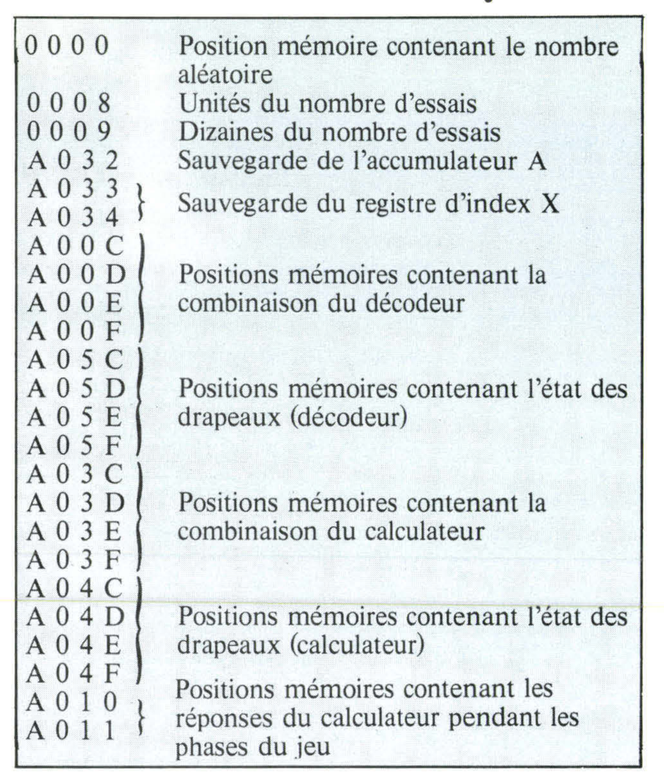



*Fig. 3 a. – $A_2 = B_2$. En comparant les positions de même rang, on positionne donc les drapeaux $d_2$ et $D_2$ à 1. P prend alors la valeur 1.*

*Fig. 3 b. – Après avoir comparé chaque Ai à chaque Bj pour lesquels les drapeaux étaient à 0 ; on obtient C = 2 et P = 1. Cela signifie deux chiffres corrects et mal placés et un chiffre correct et bien placé.*

Pour générer un nombre aléatoire, le micro-ordinateur incrémente la mémoire d'adresse 0000 depuis 1 jusqu'à 6, puis recommence à 1...

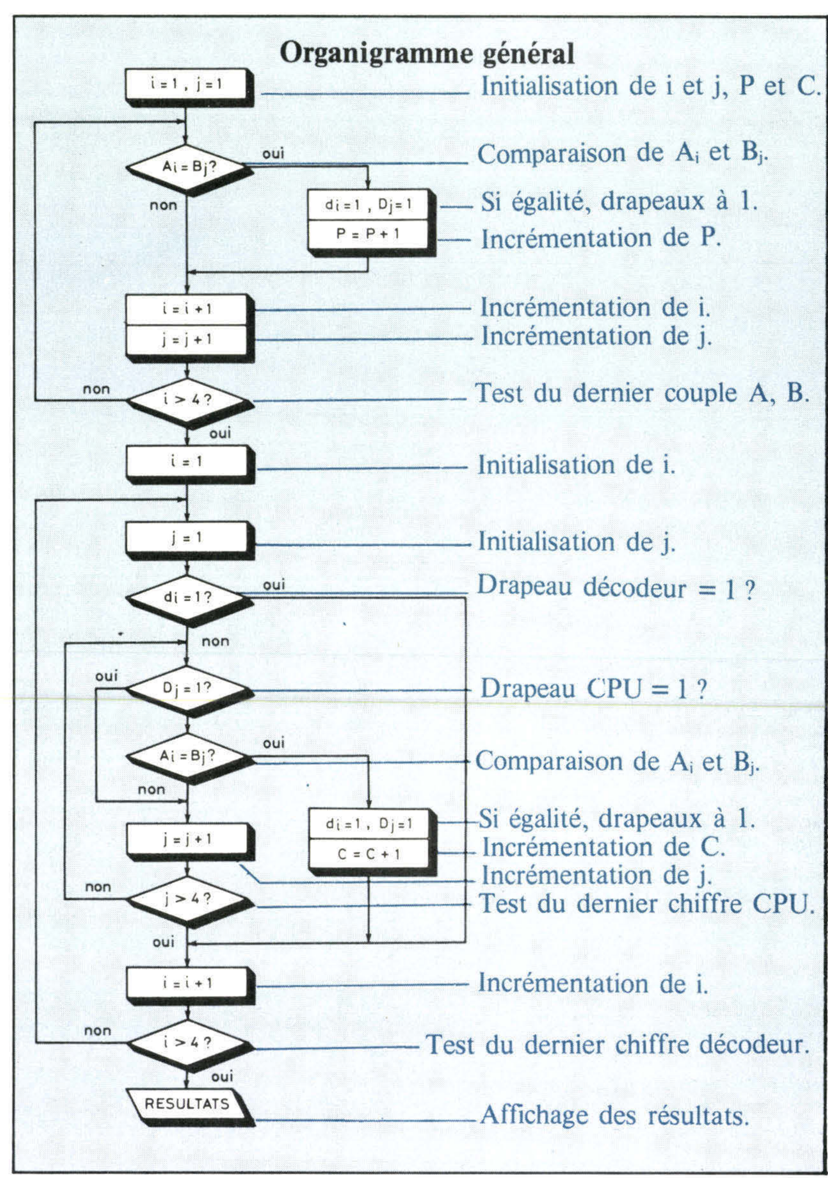
## Organisation interne des mémoires du système



| | |
|---|---|
| 0000 | Position mémoire contenant le nombre aléatoire |
| 0008 | Unités du nombre d'essais |
| 0009 | Dizaines du nombre d'essais |
| A032 | Sauvegarde de l'accumulateur A |
| A033<br>A034 | Sauvegarde du registre d'index X |
| A00C<br>A00D<br>A00E<br>A00F | Positions mémoires contenant la combinaison du décodeur |
| A05C<br>A05D<br>A05E<br>A05F | Positions mémoires contenant l'état des drapeaux (décodeur) |
| A03C<br>A03D<br>A03E<br>A03F | Positions mémoires contenant la combinaison du calculateur |
| A04C<br>A04D<br>A04E<br>A04F | Positions mémoires contenant l'état des drapeaux (calculateur) |
| A010<br>A011 | Positions mémoires contenant les réponses du calculateur pendant les phases du jeu |

Sous la présentation précédente, la configuration est la suivante :

| DRAPEAUX | JOUEUR | CPU | DRAPEAUX |
|---|---|---|---|
| A05C<br>A05D<br>A05E<br>A05F | A00C<br>A00D<br>A00E<br>A00F | A03C<br>A03D<br>A03E<br>A03F | A04C<br>A04D<br>A04E<br>A04F |




Les positions mémoires ont été choisies pour faciliter l'utilisation de l'adressage indexé ; en effet :

- A00C + 50 = A05C, etc.
- A03C + 10 = A04C, etc.

## Etude détaillée du programme « Jeu de Master Mind »

Le programme principal **(fig. 4)** fait appel à différents sous-programmes que nous allons étudier maintenant.

### Sous-programme : génération d'un nombre aléatoire

Il n'est pas question ici de générer un nombre aléatoire par des formules mathématiques compliquées. Le principe est beaucoup plus simple et donne totalement satisfaction à l'usage.

Le calculateur incrémente la mémoire 0000 depuis 1 jusqu'à 6, puis recommence à 1, etc., très rapidement. On peut considérer que le nombre contenu dans 0000 à un instant quelconque est aléatoire.

On trouvera l'organigramme et le sous-programme à la **figure 5.**

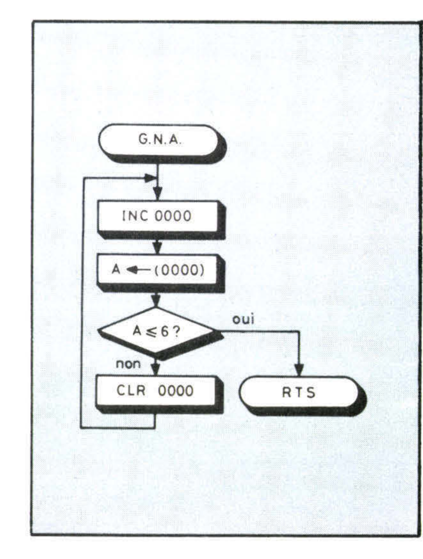
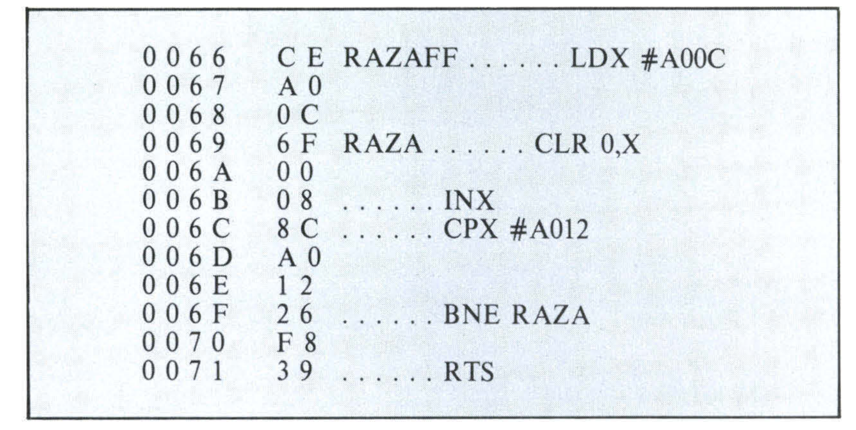



*Fig. 5. – Sous-programme de génération d'un nombre aléatoire.*

```
0088  7C  GNA ...... INC 0000
0089  00
008A  00
008B  B6  ...... LDA A ← (0000)
008C  00
008D  00
008E  81  ...... CMP A#$06
008F  06
0090  23  ...... BLS FINGNA
0091  05
0092  7F  ...... CLR 0000
0093  00
0094  00
0095  20  ...... BRA GNA
0096  F1
0097  39  FINGNA ...... RTS
```

### Sous-programmes de R à Z des mémoires d'affichage, de comptage et des différents drapeaux

Ces sous-programmes ne présentent aucune difficulté particulière, on les donne **figure 6 a, b et c,** accompagnés de leurs organigrammes.



```
0066  CE  RAZAFF ...... LDX #A00C
0067  A0
0068  0C
0069  6F  RAZA ...... CLR 0,X
006A  00
006B  08  ...... INX
006C  8C  ...... CPX #A012
006D  A0
006E  12
006F  26  ...... BNE RAZA
0070  F8
0071  39  ...... RTS
```

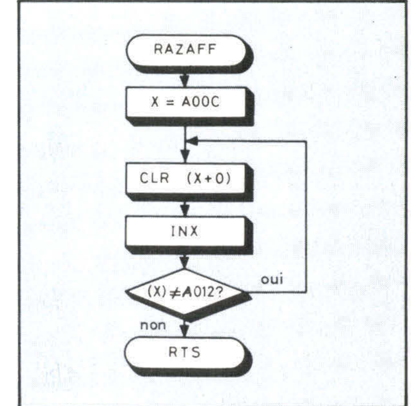
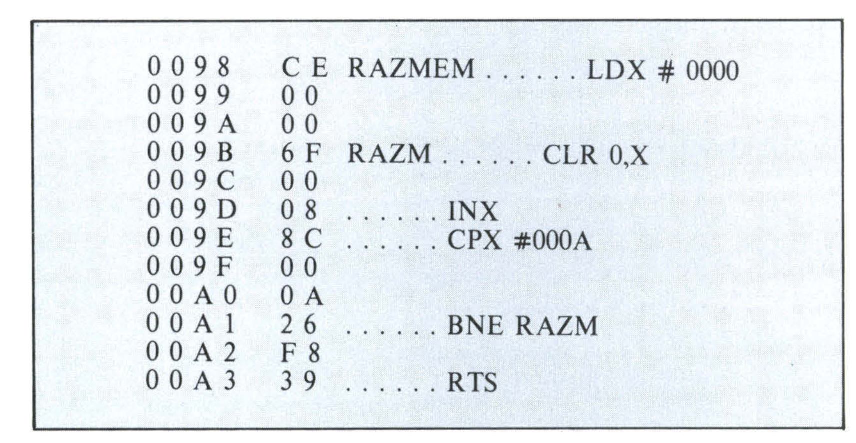



*Fig. 6 a. – Sous-programme de remise à zéro de l'affichage. Ce sous-programme met à zéro les cases mémoires d'adresses A00C à A011 (cases mémoire dont les contenus représentent les données à afficher).*



```
0098  CE  RAZMEM ...... LDX # 0000
0099  00
009A  00
009B  6F  RAZM ...... CLR 0,X
009C  00
009D  08  ...... INX
009E  8C  ...... CPX #000A
009F  00
00A0  0A
00A1  26  ...... BNE RAZM
00A2  F8
00A3  39  ...... RTS
```

RAZMEM
X = 0000
CLR (X+0)
INX
(X)≠000A?
oui
non
RTS

*Fig. 6 b. – Sous-programme de remise à zéro des mémoires de comptage. Le contenu des cases mémoire 0000 à 0009 est remis à 0 (cases mémoires contenant le nombre aléatoire et le nombre d'essais).*

```
0072  CE  RAZDRA ...... LDX #A04C
0073  A0
0074  4C
0075  6F  RAZD ...... CLR 0,X
0076  00
0077  08  ...... INX
0078  8C  ...... CPX #A060
0079  A0
007A  60
007B  26  ...... BNE RAZD
007C  F8
007D  39  ...... RTS
```

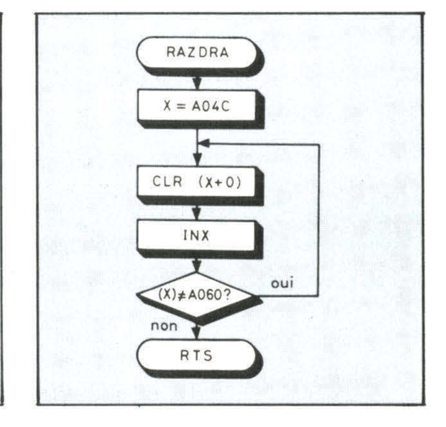



*Fig. 6 c. – Sous-programme de remise à zéro des drapeaux. Le contenu des cases mémoires d'adresses A04C à A05F sont mis à zéro.*

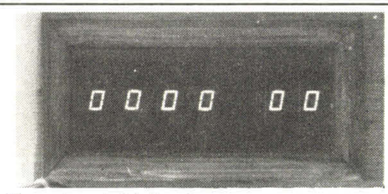


*Photo 1. - Initialisation.*

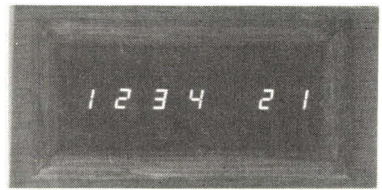


*Photo 2. - Première combinaison proposée et réponse du micro-ordinateur : 2.1.*

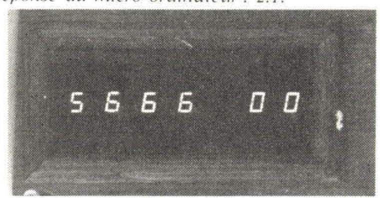


*Photo 3. - Deuxième combinaison proposée. Réponse : 0.0.*

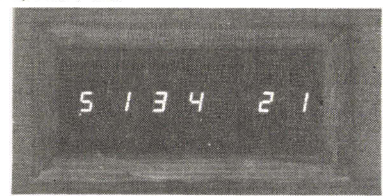


*Photo 4. - Troisième combinaison proposée. Réponse : 2.1.*

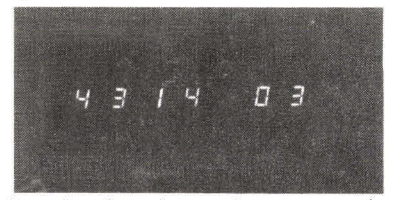


*Photo 5. - Quatrième combinaison proposée. Réponse : 0.3.*

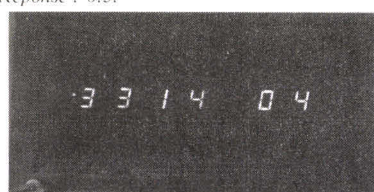


*Photo 6. - Cinquième combinaison proposée. Réponse : 0.4 (celle à trouver).*

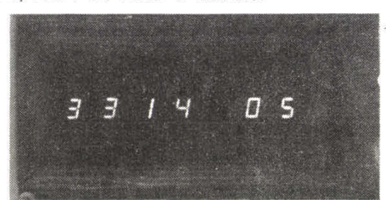


*Photo 7. - La bonne combinaison est toujours affichée, il a fallu au joueur cinq essais pour la découvrir.*

**Sous-programme « affichage »**

Ce sous-programme est identique à celui utilisé par le J-BUG, de Motorola ; nous lui avons simplement ajouté la sauvegarde de l'accumulateur A et du registre d'index X. Il est regrettable qu'on ne puisse pas utiliser directement le J-BUG, cela nous aurait libéré quelques positions mémoires.

La valeur entre parenthèses indique que le branchement s'effectue à la ligne d'adresse décimale : 170.

## Déroulement du jeu

Après avoir introduit le programme dans les mémoires RAM

```
0 0 1 F   B 7  AFFICH ...... STA A → A032
0 0 2 0   A 0
0 0 2 1   3 2
0 0 2 2   F F  ...... STX → A033, A034
0 0 2 3   A 0
0 0 2 4   3 3
0 0 2 5   C E  ...... LDX #A00C
. . . .   . .  ...............
. . . .   . .  ...............} J-BUG
0 0 5 C   F E  ...... LDX ← (A033) (A034)
0 0 5 D   A 0
0 0 5 E   3 3
0 0 5 F   B 6  ...... LDA A ← (A032)
0 0 6 0   A 0
0 0 6 1   3 2
0 0 6 2   3 9  ...... RTS
```

**Sous-programme « affichage clignotant »**

Ce sous-programme nous permet d'obtenir un affichage durant un temps déterminé par le contenu du registre x.

du calculateur et branché la planche de boutons poussoirs sur le port 8006 du PIA, la mise en route s'effectue comme cela a été indiqué ci-dessus.

Exemple de déroulement de jeu :

```
0 0 7 E   C E  AFFCLI ...... LDX #$00FF
0 0 7 F   0 0
0 0 8 0   F F
0 0 8 1   B D  AFFC ...... JSR AFFICH
0 0 8 2   0 0
0 0 8 3   1 F
0 0 8 4   0 9  ...... DEX
0 0 8 5   2 6  ...... BNE AFFC
0 0 8 6   F A
0 0 8 7   3 9  ...... RTS
```

## Programme

Le programme complet est donné sur le listing de la **figure 7.** Les déplacements des différents branchements, conditionnels ou non, sont donnés en valeur hexadécimale, mais les adresses d'arrivée sont précisées en valeur décimale.

Exemple :
179 00B3 BRA 20
180 00B4 (170) F5

– combinaisons proposées par le joueur :
** 1.2.3.4
** 5.6.6.6
** 5.1.3.4
** 4.3.1.4
** 3.3.1.4
réponses du calculateur
2 . 1
0 . 0
2 . 1
0 . 3
4 . . . . . 5 essais ■

**Max ARISTOTE**

*Fig. 7. – Listing complet du programme de jeu du « Master Mind ».*

Adresse décimale ↓ Adresse $000H

INITIALISATION

| D | H | C | M N |
|---|---|---|---|
| 00 | 00 | ? | Nb Al. |
| | 01 | | |
| 02 | 02 | | |
| 03 | 03 | | |
| | 04 | | |
| 05 | 05 | | |
| | 06 | | |
| 07 | 07 | | |
| 08 | 08 } | | Nb |
| | 09 } | | d'essais |
| | | | |
| 10 | 0A | 8E | LDS |
| | 0B | 01 | |
| 12 | 0C | FF | |
| 13 | 0D | 86 | LDAA |
| | 0E | 00 | |
| 15 | 0F | B7 | STAA |
| | 10 | 80 | |
| 17 | 11 | 06 | |
| 18 | 12 | 86 | LDAA |
| | 13 | 04 | |
| 20 | 14 | B7 | STA A |
| | 15 | 80 | |
| 22 | 16 | 07 | |
| 23 | 17 | BD | JSR |
| | 18 | 00 | |
| 25 | 19 | 66 | |
| | 1A | 7E | JMP |
| 27 | 1B | 00 | |
| 28 | 1C | A4 | |
| | 1D | 01 | NOP |
| 30 | 1E | 01 | NOP |
| | | | |
| | 1F | B7 | STA A |
| 32 | 20 | A0 | |
| 33 | 21 | 32 | |
| | 22 | FF | STX |
| 35 | 23 | A0 | |
| | 24 | 33 | |
| 37 | 25 | CE | LDX |
| 38 | 26 | A0 | |
| | 27 | 0C | |
| 40 | 28 | A6 | |
| | 29 | 00 | |
| 42 | 2A | 4C | |
| 43 | 2B | 08 | |
| | 2C | FF | |
| 45 | 2D | A0 | |
| | 2E | 20 | |
| 47 | 2F | CE | |
| 48 | 30 | E3 | |
| | 31 | C9 | |
| 50 | 32 | 08 | |
| | 33 | 4A | |
| 52 | 34 | 26 | |
| 53 | 35 | FC | |
| | 36 | 7F | |
| 55 | 37 | 80 | |
| | 38 | 22 | |
| 57 | 39 | A6 | |
| 58 | 3A | 00 | |
| | 3B | B7 | |
| 60 | 3C | 80 | |

| D | H | C | M N |
|---|---|---|---|
| | 3D | 20 | |
| 62 | 3E | B6 | |
| 63 | 3F | A0 | |
| | 40 | 1C | |
| 65 | 41 | B7 | |
| | 42 | 80 | |
| 67 | 43 | 22 | |
| 68 | 44 | CE | |
| | 45 | 00 | |
| 70 | 46 | 4D | |
| | 47 | 09 | |
| 72 | 48 | 26 | |
| 73 | 49 | FD | |
| | 4A | FE | |
| 75 | 4B | A0 | |
| | 4C | 20 | |
| 77 | 4D | 8C | |
| 78 | 4E | A0 | |
| | 4F | 12 | |
| 80 | 50 | 27 | |
| | 51 | 05 | |
| 82 | 52 | 74 | |
| 83 | 53 | A0 | |
| | 54 | 1C | |
| 85 | 55 | 20 | |
| | 56 | D1 | |
| 87 | 57 | 86 | |
| 88 | 58 | 20 | |
| | 59 | B7 | |
| 90 | 5A | A0 | |
| | 5B | 1C | |
| 92 | 5C | FE | LDX |
| 93 | 5D | A0 | |
| | 5E | 33 | |
| 95 | 5F | B6 | LDA A |
| | 60 | A0 | |
| 97 | 61 | 32 | |
| 98 | 62 | 39 | RTS |
| | 63 | 01 | NOP |
| | | | |
| 100 | 64 | 01 | NOP |
| | 65 | 01 | NOP |
| 102 | 66 | CE | LDX |
| 103 | 67 | A0 | |
| | 68 | 0C | |
| 105 | 69 | 6F | CLR |
| 106 | 6A | 00 | +X |
| | 6B | 08 | INX |
| 108 | 6C | 8C | CPX |
| | 6D | A0 | |
| 110 | 6E | 12 | |
| | 6F | 26 | BNE |
| 112 | 70 | F8 | |
| 113 | 71 | 39 | RTS |
| | 72 | CE | LDX |
| 115 | 73 | A0 | |
| | 74 | 4C | |
| 117 | 75 | 6F | CLR |
| 118 | 76 | 00 | +X |
| | 77 | 08 | INX |
| 120 | 78 | 8C | CPX |
| | 79 | A0 | |
| 122 | 7A | 60 | |
| 123 | 7B | 26 | BNE |
| | 7C | F8 | |
| 125 | 7D | 39 | RTS |

| D | H | C | M N |
|---|---|---|---|
| | 7E | CE | LDX |
| 127 | 7F | 03 | |
| 128 | 80 | FF | |
| | 81 | BD | JSR |
| 130 | 82 | 00 | |
| | 83 | 1F | |
| 132 | 84 | 09 | DEX |
| 133 | 85 | 26 | BNE |
| | 86 | FA | |
| 135 | 87 | 39 | RTS |
| 136 | 88 | 7C | INC |
| 137 | 89 | 00 | |
| 138 | 8A | 00 | |
| | 8B | B6 | LDA A |
| 140 | 8C | 00 | |
| | 8D | 00 | |
| 142 | 8E | 81 | CMP A |
| 143 | 8F | 06 | |
| | 90 | 23 | BLS |
| 145 | 91 | 05 | |
| | 92 | 7F | CLR |
| 147 | 93 | 00 | |
| 148 | 94 | 00 | |
| | 95 | 20 | BRA |
| 150 | 96 | F1 | (136) |
| | 97 | 39 | RTS |
| 152 | 98 | CE | LDX |
| 153 | 99 | 00 | |
| | 9A | 00 | |
| 155 | 9B | 6F | CLR |
| | 9C | 00 | +X |
| 157 | 9D | 08 | INX |
| 158 | 9E | 8C | CPX |
| 159 | 9F | 00 | |
| 160 | A0 | 0A | |
| | A1 | 26 | BNE |
| 162 | A2 | F8 | |
| 163 | A3 | 39 | RTS |
| | A4 | BD | JSR |
| 165 | A5 | 00 | |
| | A6 | 72 | |
| 167 | A7 | BD | JSR |
| 168 | A8 | 00 | |
| | A9 | 98 | |
| 170 | AA | BD | JSR |
| | AB | 00 | |
| 172 | AC | 1F | |
| 173 | AD | B6 | LDA A |
| | AE | 80 | |
| 175 | AF | 06 | |
| | B0 | 48 | ASL A |
| 177 | B1 | 25 | BCS |
| 178 | B2 | 02 | (181) |
| | B3 | 20 | BRA |
| 180 | B4 | F5 | (170) |
| | B5 | CE | LDX |
| 182 | B6 | A0 | |
| 183 | B7 | 3B | |
| | B8 | BD | JSR |
| 185 | B9 | 00 | |
| | BA | 88 | |
| 187 | BB | BD | JSR |
| 188 | BC | 00 | |
| | BD | 1F | |
| 190 | BE | B6 | LDA A |
| | BF | 80 | |

| D | H | C | M N |
|---|---|---|---|
| 192 | C0 | 06 | |
| 193 | C1 | 48 | ASL A |
| | C2 | 48 | ASL A |
| 195 | C3 | 25 | BCS |
| | C4 | 02 | (199) |
| 197 | C5 | 20 | BRA |
| 198 | C6 | F1 | (184) |
| | C7 | 08 | INX |
| 200 | C8 | BD | JSR |
| | C9 | 01 | |
| 202 | CA | CB | |
| 203 | CB | 27 | BEQ |
| | CC | 11 | (222) |
| 205 | CD | BD | JSR |
| | CE | 00 | |
| 207 | CF | 1F | |
| 208 | D0 | B6 | LDA A |
| | D1 | 00 | |
| 210 | D2 | 00 | |
| 211 | D3 | A7 | STA A |
| 212 | D4 | 00 | +X |
| 213 | D5 | B6 | LDA A |
| | D6 | 80 | |
| 215 | D7 | 06 | |
| | D8 | 48 | ASL A |
| 217 | D9 | 48 | ASL A |
| 218 | DA | 24 | BCC |
| | DB | DC | (184) |
| 220 | DC | 20 | BRA |
| | DD | EF | (205) |
| 222 | DE | BD | JSR |
| 223 | DF | 01 | |
| | E0 | D2 | |
| 225 | E1 | 7C | INC |
| | E2 | 00 | |
| 227 | E3 | 08 | |
| 228 | E4 | B6 | LDA A |
| | E5 | 00 | |
| 230 | E6 | 08 | |
| | E7 | 81 | CMP A |
| 232 | E8 | 09 | |
| 233 | E9 | 22 | BHI |
| | EA | 02 | (237) |
| 235 | EB | 20 | BRA |
| | EC | 17 | (260) |
| 237 | ED | 7F | CLR |
| 238 | EE | 00 | |
| | EF | 08 | |
| 240 | F0 | 7C | INC |
| | F1 | 00 | |
| 242 | F2 | 09 | |
| 243 | F3 | B6 | LDA A |
| | F4 | 00 | |
| 245 | F5 | 09 | |
| | F6 | 81 | CMP A |
| 247 | F7 | 09 | |
| 248 | F8 | 22 | BHI |
| | F9 | 02 | (252) |
| 250 | FA | 20 | BRA |
| | FB | 08 | (260) |
| 252 | FC | 7F | CLR |
| 253 | FD | 00 | |
| | FE | 08 | |
| 255 | FF | 7F | CLR |

Ajouter 256 à l'adresse décimale

Adresse 0100H

| D | H | C | M N |
|---|---|---|---|
| 00 | 00 | 00 | |
| | 01 | 09 | |
| 02 | 02 | 20 | BRA |
| 03 | 03 | DD | (225) |
| | 04 | B6 | LDA A |
| 05 | 05 | 80 | (PIA) |
| | 06 | 06 | |
| 07 | 07 | 5F | CLR B |
| 08 | 08 | BD | JSR |
| | 09 | 00 | |
| | | | |
| 10 | 0A | 1F | |
| | 0B | 5C | INC B |
| 12 | 0C | C1 | CMP B |
| 13 | 0D | 06 | |
| | 0E | 22 | BHI |
| 15 | 0F | F4 | (260) |
| | 10 | 46 | ROR A |
| 17 | 11 | 25 | BCS |
| 18 | 12 | 02 | (277) |
| | 13 | 20 | BRA |
| 20 | 14 | F3 | (264) |
| | 15 | E7 | STA B |
| 22 | 16 | 00 | +X |
| 23 | 17 | 08 | INX |
| | 18 | 8C | CPX |
| 25 | 19 | A0 | |
| | 1A | 10 | |
| 27 | 1B | 27 | BEQ |
| 28 | 1C | 0F | (300) |
| | 1D | BD | JSR |
| 30 | 1E | 00 | |
| | 1F | 1F | |
| 32 | 20 | B6 | LDA A |
| 33 | 21 | 80 | (PIA) |
| | 22 | 06 | |
| 35 | 23 | 81 | CMP A |
| | 24 | 00 | |
| 37 | 25 | 27 | BEQ |
| 38 | 26 | 02 | (297) |
| | 27 | 20 | BRA |
| 40 | 28 | F4 | (285) |
| | 29 | 7E | JMP |
| 42 | 2A | 01 | |
| 43 | 2B | 04 | |
| | 2C | BD | JSR |
| 45 | 2D | 00 | |
| | 2E | 1F | |
| 47 | 2F | B6 | LDA A |
| 48 | 30 | 80 | |
| | 31 | 06 | |
| 50 | 32 | 48 | ASL A |
| | 33 | 48 | ASL A |
| 52 | 34 | 25 | BCS |
| 53 | 35 | 02 | (312) |
| | 36 | 20 | BRA |
| 55 | 37 | F4 | (300) |
| | 38 | CE | LDX |
| 57 | 39 | A0 | |
| 58 | 3A | 0B | |
| | 3B | 08 | INX |
| 60 | 3C | E6 | LDA B |
| | 3D | 00 | +X |

| D | H | C | M N |
|---|---|---|---|
| 62 | 3E | A6 | LDA A |
| 63 | 3F | 30 | +X |
| | 40 | 11 | CBA |
| 65 | 41 | 27 | BEQ |
| | 42 | 02 | (325) |
| 67 | 43 | 20 | BRA |
| 68 | 44 | 07 | (332) |
| | 45 | 6C | INC |
| 70 | 46 | 40 | +X |
| | 47 | 6C | INC |
| 72 | 48 | 50 | +X |
| 73 | 49 | 7C | INC |
| | 4A | A0 | |
| 75 | 4B | 11 | |
| | 4C | 8C | CPX |
| 77 | 4D | A0 | |
| 78 | 4E | 0F | |
| | 4F | 27 | BEQ |
| 80 | 50 | 02 | (339) |
| | 51 | 20 | BRA |
| 82 | 52 | E8 | (315) |
| 83 | 53 | CE | LDX |
| | 54 | A0 | |
| 85 | 55 | 3B | |
| | 56 | 08 | INX 1 |
| 87 | 57 | A6 | LDA A |
| 88 | 58 | 10 | +X |
| | 59 | 81 | CMP A |
| 90 | 5A | 00 | |
| | 5B | 27 | BEQ |
| 92 | 5C | 02 | (351) |
| 93 | 5D | 20 | BRA |
| | 5E | 29 | (392) |
| 95 | 5F | FF | STX 1 |
| | 60 | A0 | |
| 97 | 61 | 33 | |
| 98 | 62 | A6 | LDA A |
| | 63 | 00 | +X1 |
| | | | |
| 100 | 64 | CE | LDX 2 |
| | 65 | A0 | |
| 102 | 66 | 0B | |
| 103 | 67 | 08 | INX 2 |
| | 68 | E6 | LDA B |
| 105 | 69 | 50 | +X2 |
| 106 | 6A | C1 | CMP B |
| | 6B | 00 | |
| 108 | 6C | 27 | BEQ |
| | 6D | 02 | (368) |
| 110 | 6E | 20 | BRA |
| | 6F | 0E | (382) |
| 112 | 70 | E6 | LDA B |
| 113 | 71 | 00 | +X2 |
| | 72 | 11 | CBA |
| 115 | 73 | 27 | BEQ |
| | 74 | 02 | (375) |
| 117 | 75 | 20 | BRA |
| 118 | 76 | 07 | (382) |
| | 77 | 6C | INC |
| 120 | 78 | 50 | +X2 |
| | 79 | 7C | INC |
| 122 | 7A | A0 | |
| 123 | 7B | 10 | |
| | 7C | 20 | BRA |
| 125 | 7D | 07 | (389) |

| D | H | C | M N |
|---|---|---|---|
| | 7E | 8C | CPX 2 |
| 127 | 7F | A0 | |
| 128 | 80 | 0F | |
| | 81 | 27 | BEQ |
| 130 | 82 | 02 | (389) |
| | 83 | 20 | BRA |
| 132 | 84 | E2 | (359) |
| 133 | 85 | FE | LDX 1 |
| | 86 | A0 | |
| 135 | 87 | 33 | |
| | 88 | 8C | CPX 1 |
| 137 | 89 | A0 | |
| 138 | 8A | 3F | |
| | 8B | 27 | BEQ |
| 140 | 8C | 02 | (399) |
| | 8D | 20 | BRA |
| 142 | 8E | C7 | (342) |
| 143 | 8F | BD | JSR |
| | 90 | 00 | |
| 145 | 91 | 72 | |
| | 92 | BD | JSR |
| 147 | 93 | 00 | |
| 148 | 94 | 1F | |
| | 95 | B6 | LDA A |
| 150 | 96 | A0 | |
| | 97 | 11 | |
| 152 | 98 | 81 | CMP A |
| 153 | 99 | 04 | |
| | 9A | 27 | BEQ |
| 155 | 9B | 21 | (445) |
| | 9C | B6 | LDA A |
| 157 | 9D | 80 | |
| 158 | 9E | 06 | |
| 159 | 9F | 48 | ASL A |
| 160 | A0 | 48 | ASL A |
| | A1 | 24 | BCC |
| 162 | A2 | 05 | (424) |
| 163 | A3 | 20 | BRA |
| | A4 | ED | (402) |
| 165 | A5 | 01 | NOP |
| | A6 | 01 | NOP |
| 167 | A7 | 01 | NOP |
| 168 | A8 | BD | JSR |
| | A9 | 00 | |
| 170 | AA | 66 | |
| | AB | 7E | JMP |
| 172 | AC | 00 | |
| 173 | AD | DE | |
| | AE | B6 | LDA A |
| 175 | AF | 00 | |
| | B0 | 08 | |
| 177 | B1 | B7 | STA A |
| 178 | B2 | A0 | |
| | B3 | 11 | |
| 180 | B4 | B6 | LDA A |
| | B5 | 00 | |
| 182 | B6 | 09 | |
| 183 | B7 | B7 | STA A |
| | B8 | A0 | |
| 185 | B9 | 10 | |
| | BA | BD | JSR |
| 187 | BB | 00 | |
| 188 | BC | 7E | |
| | BD | 7F | CLR |
| 190 | BE | A0 | |

| D | H | C | M N |
|---|---|---|---|
| | BF | 10 | |
| 192 | C0 | 86 | LDA A |
| 193 | C1 | 04 | |
| | C2 | B7 | STA A |
| 195 | C3 | A0 | |
| | C4 | 11 | |
| 197 | C5 | BD | JSR |
| 198 | C6 | 00 | |
| | C7 | 7E | |
| 200 | C8 | 7E | JMP |
| | C9 | 01 | |
| 202 | CA | D9 | |
| 203 | CB | 7C | INC |
| | CC | A0 | |
| 205 | CD | 11 | |
| | CE | 8C | CPX |
| 207 | CF | A0 | |
| 208 | D0 | 40 | |
| | D1 | 39 | RTS |
| 210 | D2 | 7F | CLR |
| 211 | D3 | A0 | |
| 212 | D4 | 11 | |
| 213 | D5 | CE | LDX |
| | D6 | A0 | |
| 215 | D7 | 0C | |
| | D8 | 39 | RTS |
| 217 | D9 | B6 | LDA A |
| 218 | DA | 80 | (PIA) |
| | DB | 06 | |
| 220 | DC | 48 | ASL A |
| | DD | 48 | ASL A |
| 222 | DE | 24 | BCC |
| 223 | DF | 03 | (483) |
| | E0 | 7E | JMP |
| 225 | E1 | 01 | |
| | E2 | AE | |
| 227 | E3 | 7E | JMP |
| 228 | E4 | 00 | |
| | E5 | 17 | |
| 230 | E6 | | FIN |
| | E7 | | |
| 232 | E8 | | |
| 233 | E9 | | |
| | EA | | |
| 235 | EB | | |
| | EC | | |
| 237 | ED | | |
| 238 | EE | | |
| | EF | | |
| 240 | F0 | | |
| | F1 | | |
| 242 | F2 | | |
| 243 | F3 | | |
| | F4 | | |
| 245 | F5 | | |
| | F6 | | |
| 247 | F7 | | |
| 248 | F8 | | |
| | F9 | | |
| 250 | FA | | |
| | FB | | |
| 252 | FC | | |
| 253 | FD | | |
| | FE | | |
| 255 | FF | | PILE |

# La puissance du Z 80 utilisée dans le terminal Mostek à disques souples.

## Système de développement SYS-80FT

Le SYS-80FT Mostek est un appareil puissant pour développer des programmes d'applications microprocesseur Z80. Il comprend en effet tout le logiciel et le matériel nécessaire. Le SYS-80FT comporte quatre cartes (format Eurocarte double largeur):

- OEM-80E, avec CPU + 16 K octets RAM + E/S
- RAM-80E, pour l'extension des capacités mémoire et E/S
- FLP-80E, pour la commande du système à double disque
- VDI-P/S, pour l'interface clavier-écran.

ainsi qu'un interface interne pour les accessoires périphériques tels qu'imprimante, ruban perforé, programmeur de PROM, etc... De plus il y a trois emplacements de cartes laissés libres pour une extension mémoire et E/S éventuelle.

En option, la carte AIM-80E permet l'émulation complète en temps réel et l'accessoire PPG-08/16 sert à la programmation des mémoires PROM. Le SYS-80FT est donc l'outil de base dont vous avez besoin pour concevoir vos applications.

## Micro-ordinateur

Le logiciel du SYS-80FT est très complet. Avec les nouveaux programmes Basic et Fortran maintenant disponibles, il peut aisément s'utiliser en micro-ordinateur d'usage général.
Le SYS-80FT Mostek est un système intégré dans un coffret; les cartes sont disponibles séparément pour les applications OEM.
Pour en savoir plus, écrivez ou téléphonez à Mostek.




Salon des composants :
Allée 13 - Stand 164

**MOSTEK®**

Microprocesseurs et Systèmes de développement

Mostek fournit également: Circuits Mémoires, Circuits Télécommunications et Cartes-Mémoires.

**Mostek France SARL,** 30, rue du Morvan, Silic 505 - 94623 RUNGIS Cedex.
Tél.: (1) 687 34 14
Distributeurs:
**S.C.A.I.B. SA,** 80 rue d'Arcueil, 94150 RUNGIS Tél: (1) 687 23 13
**P E P,** 45 rue Barthélémy, 92120 MONTROUGE Tél : (1) 735 32 20
Pour la Belgique: **SOTRONIC,** 14 rue Père De Deken, B-1040 BRUXELLES. Tél: 02/736 1007
Pour la Suisse: **MEMOTEC,** CH-4932 LOTZWIL (Berne) Tél : 063/28 11 22
Pour les autres pays européens: **Mostek International,** Bruxelles. Tél: 02/660 25 68

# Le BASIC

**Jusqu'à présent, nous nous sommes quelque peu cantonnés dans les calculs, les définitions et les impressions graphiques.**

**Voici enfin venue l'heure des gestionnaires. En réalité il s'agit des petits gestionnaires, ayant à traiter des fichiers de 100, 200, voire 300 postes. Pour traiter un nombre supérieur d'éléments, la programmation en assembleur ou dans des langages spécialement étudiés et orientés vers la gestion s'imposera.**

**Néanmoins, même s'il s'adresse aux « Beginners », le BASIC permet de gérer et de trier en particulier des fichiers. Cela grâce à des instructions de traitement de chaînes de caractères comme STRING, VAL, STR$, LEN, ASC, MID$, CHR$, etc.**

**Pour imprimer une suite de lettres, circulaires vers un fichier d'amis ou de clients, ou pour amorcer automatiquement l'impression d'un bon de commande de réapprovisionnement en cas d'atteinte du stock-limite d'un produit, ces instructions sont essentielles.**

**Les éléments et critères relatifs aux fichiers tels que la taille, les supports et les modes d'organisation sont présentés en encadré.**

## « SNOOPY »

*Extrait de la revue américaine « Creative Computing », le chien Snoopy programmé en Basic.*

Commençons par un exemple pratique : envoyer une suite de cartes de vœux, imprimées par le micro-ordinateur sur une machine à écrire connectable en ASCII ou par un mécanisme à relais géré par le système lui-même. Nous disposons d'une liste de noms et d'adresses d'amis, qui devra être imprimée selon des règles de mise en page établies en fonction de nos goûts, soit sur des cartes toutes faites soit accompagnées d'un texte.

Comment résoudre ce problème ? Une solution relativement intéressante à première vue serait d'installer chaque étiquette dans une instruction PRINT du type :

```
N° ligne 1     « PRINT » . . . . M. Georges Kournonsky
                         . . . . 23, bd des Batignolles
                         . . . . 75678 PARIS
                         . . . . . . . . . . . . . . . . .
N° ligne 2     « PRINT »   une autre adresse...
```

Malheureusement, ce genre de programmation demande la réécriture de toute la zone à imprimer, même si l'on veut modifier une particule (remplacer M. par M. et Mme, par exemple). D'autre part la place mémoire sera largement pourvue de blancs, de retours à la ligne, etc., nécessaires au graphisme de chaque étiquette. Le plus grave, cependant, reste l'impossibilité de traiter le fichier-amis. Imaginons que l'on veuille envoyer une lettre aux personnes habitant la région parisienne. Il nous faut pouvoir extraire le mot « PARIS » ou « BANLIEUE » des étiquettes et n'imprimer que ces lettres.

Pour obtenir cela on utilise des instructions spécifiques aux chaînes de caractères.

## Variables chaînes de caractères

Une chaîne de caractères est un ensemble de caractères placés les uns à la suite des autres.

A la différence des variables numériques, qui s'écrivent sous

Une chaîne de caractères est un ensemble de caractères placés les uns à la suite des autres.

forme d'une lettre suivie ou non d'un chiffre, les chaînes de caractères sont nommées par une lettre suivie du signe dollar, $ : **A$, B$,** ... par exemple.

L'affectation d'une valeur à une telle variable utilise la mise entre guillemets de la chaîne de caractères à nommer :

A$ = « ABCD    PAUL    DURAND 123 F »

Ces variables respectent une règle d'additivité *. Ainsi, si A$ = « ABCD » et B$ = « EFGH », la variable

Y$ = A$ + B$

sera égale à

ABCDEFGH.

Cette variable, rencontrée pour la première fois (ou à initialiser), est vide, ce qui s'exprime par un blanc entre guillemets :

C$ = « »

Les guillemets qui délimitent une chaîne de caractères lors de sa déclaration, n'apparaissent plus lors de l'impression :

PRINT A$ + A$

donnera : ABCDABCD, sans guillemets.

De même, la lecture des chaînes de caractères en DATA se passe de guillemets :

```
READ A$, B$
.
.
.
.
DATA ABCD, EFGH
```

et leur entrée en INPUT aussi :

```
INPUT A$
? ABCD (CR)
```

Lors de l'exécution du programme (RUN) la machine demande l'entrée de A$ par (?), la touche retour de chariot (CR : carriage return) donnant la suite du programme.

## Instructions portant sur les chaînes de caractères

**STRING**

Cette instruction définit la longueur des variables-caractères. Compte tenu de la taille-mémoire et des machines, la dimension par défaut d'une déclaration de longueur (STRING) varie de 10 à 20 caractères.

Si on déclare : STRING = 128, toutes les variables-caractère qui suivront la ligne du STRING auront 128 octets réservés en mémoire (1 octet = 1 caractère).

**STR$ (x)**

Où x est une variable numérique, une constante ou une expression arithmétique, est l'instruction qui permet de transformer des valeurs numériques en chaînes de caractères de même écriture.

En effet : 12, chiffre décimal et 12 en code ASCII, visualisé ou imprimé, ne sont pas les mêmes, car, dans le deuxième cas, il s'agit de la représentation symbolique d'un 1 et d'un 2 côte à côte.

**Exemple :** si A = 12 et Y$ = STR$ (A), Y$ représentera pour la suite du programme la chaîne de caractères Y$ = « 1,2 ».

**CHR$ (x)**

(x ayant la même signification que précédemment) fabrique un caractère, correspondant à la valeur décimale x en ASCII.

**Exemple :** si A = 65, CHR$ (A) sera le caractère A ; de même CHR$ (66) = B.

De la même manière, le programme suivant permet de visualiser tout le code ASCII :

```
10 FOR I = 0 TO 255
20 A$ = CHR$ (I)
30 PRINT I, A$ (ou CHR$ (I) directement)
40 NEXT I
50 END
```

**MID$**

Cette fonction permet de visualiser un ou plusieurs caractères d'une chaîne. Elle constitue une variable-caractère à son tour et s'écrit sous la forme

MID$ (Var$, I, J)

A partir du I-ème caractère de la chaîne Var$, en commençant par la gauche (le premier caractère ayant pour matricule I = 1 et non pas 0), on prend J caractères pour former la nouvelle chaîne appelée MID$.

**Exemples :**

```
A$ = « 0123456789ABCDEF »
Y$ = MID$ (A$, 1, 1) sera alors « 0 »
     MID$ (A$, 3, 1)            « 2 »
     MID$ (A$, 3, 2)            « 23 »
     MID$ (A$, 16, 2)           « F »
```

L'intérêt évident d'une telle fonction est de nous permettre d'extraire un mot (« PARIS » de l'exemple d'une chaîne de caractères pouvant constituer une adresse d'ouverture) et de le comparer à un autre mot, de l'additionner, etc.

**LEFT$**

Fonction similaire à la précédente mais on prend comme point de départ I, toujours le caractère le plus à gauche. Un, deux, ou plusieurs caractères à partir de la gauche pourront ainsi être détachés du reste de la chaîne de caractères pour former une autre chaîne :

```
LEFT$ (A$, 2) sera égal à « 01 »
LEFT$ (A$, 6) sera égal à « 012345 »
```

**RIGHT $**

Même fonction que la précédente avec le départ (du comptage) à partir de la droite :

```
RIGHT $ (A$, 2) sera égal à « EF »
RIGHT $ (A$, 6)             « ABCDEF »
```

Remarquons l'intérêt de ces fonctions dans la recherche de codes dans un fichier.

Reprenons notre problème ini-

* *Cette opération porte le nom de « concaténation » et permet de mettre bout à bout deux chaînes de caractères afin d'en constituer une troisième.*

*Programme de visualisation du code ASCII. Pour I variant de 1 à 256, l'instruction CHR$ (I) traduit la valeur décimale de I par le caractère du code ASCII correspondant. Cette classification des caractères permet d'établir une relation d'ordre entre chacun d'eux.*

```
0010 FOR I = 1 TO 256
0020 PRINT "VAL.DECIMALE = ";I;"   ";"VAL.ASCII = ";CHR$(I)
0030 NEXT I
0040 END


READY
#RUN
VAL.DECIMALE =  1    VAL.ASCII =
VAL.DECIMALE =  2    VAL.ASCII =
VAL.DECIMALE =  3    VAL.ASCII =
VAL.DECIMALE =  4    VAL.ASCII =
VAL.DECIMALE =  5    VAL.ASCII =
VAL.DECIMALE =  6    VAL.ASCII =
VAL.DECIMALE =  7    VAL.ASCII =
VAL.DECIMALE =  8    VAL.ASCII =
VAL.DECIMALE =  9    VAL.ASCII =
VAL.DECIMALE =  10    VAL.ASCII =

VAL.DECIMALE =  11    VAL.ASCII =
VAL.DECIMALE =  12    VAL.ASCII =
VAL.DECIMALE =  13    VAL.ASCII =
VAL.DECIMALE =  14    VAL.ASCII =
VAL.DECIMALE =  15    VAL.ASCII =
VAL.DECIMALE =  16    VAL.ASCII =
VAL.DECIMALE =  17    VAL.ASCII =
VAL.DECIMALE =  18    VAL.ASCII =
VAL.DECIMALE =  19    VAL.ASCII =
VAL.DECIMALE =  20    VAL.ASCII =
VAL.DECIMALE =  21    VAL.ASCII =
VAL.DECIMALE =  22    VAL.ASCII =
VAL.DECIMALE =  23    VAL.ASCII =
VAL.DECIMALE =  24    VAL.ASCII =
VAL.DECIMALE =  25    VAL.ASCII =
VAL.DECIMALE =  26    VAL.ASCII =
VAL.DECIMALE =  27    VAL.ASCII =
VAL.DECIMALE =  28    VAL.ASCII =
VAL.DECIMALE =  29    VAL.ASCII =
VAL.DECIMALE =  30    VAL.ASCII =
VAL.DECIMALE =  31    VAL.ASCII =
VAL.DECIMALE =  32    VAL.ASCII =
VAL.DECIMALE =  33    VAL.ASCII = !
VAL.DECIMALE =  34    VAL.ASCII = "
VAL.DECIMALE =  35    VAL.ASCII = #
VAL.DECIMALE =  36    VAL.ASCII = $
VAL.DECIMALE =  37    VAL.ASCII = %
VAL.DECIMALE =  38    VAL.ASCII = &
VAL.DECIMALE =  39    VAL.ASCII = '
VAL.DECIMALE =  40    VAL.ASCII = (
VAL.DECIMALE =  41    VAL.ASCII = )
VAL.DECIMALE =  42    VAL.ASCII = *
VAL.DECIMALE =  43    VAL.ASCII = +
VAL.DECIMALE =  44    VAL.ASCII = ,
VAL.DECIMALE =  45    VAL.ASCII = -
VAL.DECIMALE =  46    VAL.ASCII = .
VAL.DECIMALE =  47    VAL.ASCII = /
VAL.DECIMALE =  48    VAL.ASCII = 0
VAL.DECIMALE =  49    VAL.ASCII = 1
VAL.DECIMALE =  50    VAL.ASCII = 2
VAL.DECIMALE =  51    VAL.ASCII = 3
VAL.DECIMALE =  52    VAL.ASCII = 4
VAL.DECIMALE =  53    VAL.ASCII = 5
VAL.DECIMALE =  54    VAL.ASCII = 6
VAL.DECIMALE =  55    VAL.ASCII = 7
VAL.DECIMALE =  56    VAL.ASCII = 8
VAL.DECIMALE =  57    VAL.ASCII = 9
VAL.DECIMALE =  58    VAL.ASCII = :
VAL.DECIMALE =  59    VAL.ASCII = ;
VAL.DECIMALE =  60    VAL.ASCII = <
VAL.DECIMALE =  61    VAL.ASCII = =
VAL.DECIMALE =  62    VAL.ASCII = >
VAL.DECIMALE =  63    VAL.ASCII = ?
VAL.DECIMALE =  64    VAL.ASCII = @
VAL.DECIMALE =  65    VAL.ASCII = A
VAL.DECIMALE =  66    VAL.ASCII = B
VAL.DECIMALE =  67    VAL.ASCII = C
VAL.DECIMALE =  68    VAL.ASCII = D
VAL.DECIMALE =  69    VAL.ASCII = E
VAL.DECIMALE =  70    VAL.ASCII = F
VAL.DECIMALE =  71    VAL.ASCII = G
VAL.DECIMALE =  72    VAL.ASCII = H
VAL.DECIMALE =  73    VAL.ASCII = I
VAL.DECIMALE =  74    VAL.ASCII = J
VAL.DECIMALE =  75    VAL.ASCII = K
VAL.DECIMALE =  76    VAL.ASCII = L
VAL.DECIMALE =  77    VAL.ASCII = M
VAL.DECIMALE =  78    VAL.ASCII = N
VAL.DECIMALE =  79    VAL.ASCII = O
VAL.DECIMALE =  80    VAL.ASCII = P
VAL.DECIMALE =  81    VAL.ASCII = Q
VAL.DECIMALE =  82    VAL.ASCII = R
VAL.DECIMALE =  83    VAL.ASCII = S
VAL.DECIMALE =  84    VAL.ASCII = T
VAL.DECIMALE =  85    VAL.ASCII = U
VAL.DECIMALE =  86    VAL.ASCII = V
VAL.DECIMALE =  87    VAL.ASCII = W
VAL.DECIMALE =  88    VAL.ASCII = X
VAL.DECIMALE =  89    VAL.ASCII = Y
VAL.DECIMALE =  90    VAL.ASCII = Z
VAL.DECIMALE =  91    VAL.ASCII = [
VAL.DECIMALE =  92    VAL.ASCII = \
VAL.DECIMALE =  93    VAL.ASCII = ]
VAL.DECIMALE =  94    VAL.ASCII = ^
VAL.DECIMALE =  95    VAL.ASCII = _
VAL.DECIMALE =  96    VAL.ASCII = `
VAL.DECIMALE =  97    VAL.ASCII = a
VAL.DECIMALE =  98    VAL.ASCII = b
VAL.DECIMALE =  99    VAL.ASCII = c
VAL.DECIMALE =  100    VAL.ASCII = d
VAL.DECIMALE =  101    VAL.ASCII = e
VAL.DECIMALE =  102    VAL.ASCII = f
VAL.DECIMALE =  103    VAL.ASCII = g
VAL.DECIMALE =  104    VAL.ASCII = h
VAL.DECIMALE =  105    VAL.ASCII = i
VAL.DECIMALE =  106    VAL.ASCII = j
VAL.DECIMALE =  107    VAL.ASCII = k
VAL.DECIMALE =  108    VAL.ASCII = l
VAL.DECIMALE =  109    VAL.ASCII = m
VAL.DECIMALE =  110    VAL.ASCII = n
VAL.DECIMALE =  111    VAL.ASCII = o
VAL.DECIMALE =  112    VAL.ASCII = p
VAL.DECIMALE =  113    VAL.ASCII = q
VAL.DECIMALE =  114    VAL.ASCII = r
VAL.DECIMALE =  115    VAL.ASCII = s
VAL.DECIMALE =  116    VAL.ASCII = t
VAL.DECIMALE =  117    VAL.ASCII = u
VAL.DECIMALE =  118    VAL.ASCII = v
VAL.DECIMALE =  119    VAL.ASCII = w
VAL.DECIMALE =  120    VAL.ASCII = x
VAL.DECIMALE =  121    VAL.ASCII = y
VAL.DECIMALE =  122    VAL.ASCII = z
VAL.DECIMALE =  123    VAL.ASCII = {
VAL.DECIMALE =  124    VAL.ASCII = |
VAL.DECIMALE =  125    VAL.ASCII = }
VAL.DECIMALE =  126    VAL.ASCII = ~
....
```

Un fichier est un ensemble d'informations organisé de manière à pouvoir être traité par la machine à partir de différents langages évolués ou non.



tial : lettre vers des amis parisiens.

Nous avons plusieurs possibilités :

1) Les placer en DATA et les lire par un READ A$ (adresse $).

2) Les déclarer en tant que variables dans le programme, s'ils ne sont pas très nombreux.

3) Les faire entrer à partir d'un support magnétique par un INPUT (canal choisi) ou quelque routine de système à disque magnétique, etc.

Reprenons dans l'ordre. Voici un exemple illustrant cette possibité.

```
 10 X$ = « PARIS »
 30 FOR I = 1ER AMI TO DERNIER (1 TO 3)
 40 READ A$ : M$ = MID $ (A$, 15, 5)
 50 IF M$ = X$ THEN PRINT (canal imprimante) A$
 60 NEXT I
 70 DATA DURAND RUE X    PARIS    .... ETC ....
 80 DATA PAUL RUE XYZ    EVRY     .... ETC ....
 90 DATA MICHELE         PARIS    .... ETC ....
100 END.
```

Nous avons simplifié au maximum l'écriture des adresses en mettant néanmoins le nom de la ville dans une même colonne ou zone d'enregistrement.

Au lancement du programme seules les lignes 70 et 90 s'imprimeront.

L'avantage de cette méthode est de pouvoir lire parfois des supports extérieurs. Les DATA pourront être stockés en autant de lignes de programme que l'on désire, sur une minicassette. Ils pourront être chargés consécutivement, dans la limite de place mémoire de programme disponible.

Les adresses pourront être déclarées sous la forme :

```
A$ = « DURAND RUE X PARIS »
B$ = « . . . . », ETC.
```

La méthode est propice aux petites listes, car on sature très vite à 24 variables-chaînes de caractères, lesquelles, même poussées par un string à leur maximum de longueur (128 ou 256 en fonction des machines) ne pourront pas tout contenir.

Les manipulations des supports magnétiques extérieurs, autres que la minicassette-amateur exigent des ordres spécifiques qui pourront apparaître comme des particules aux instructions d'entrée/sortie existantes.

**Exemple :**

PRINT ON ; PRINT # 4 ;

INPUT @ 6 etc.

Quelle que soit la nature de ces instructions, elles aboutiront à ranger un enregistrement d'un fichier sur disque, par exemple, dans une chaîne de caractères qui lui est réservée par le programme.

Il s'agira ensuite de traiter l'enregistrement ainsi appelé en mémoire centrale et de le re-déposer sur le support magnétique mis à jour. Une annexe donnera quelques notions sur les fichiers. La diversité des machines à ce point est telle que nous ne pouvons que rester très général. Néanmoins, supposons que l'on dispose d'un fichier en mémoire, sous l'une des formes abordées précédemment. Mise à part la recherche de chaînes de caractères, il nécessitera parfois des tris. ■

**A. DORIS**

## Notions sur les fichiers

**En informatique, un fichier est un ensemble d'informations (ensemble de fiches) organisé de manière à pouvoir être traité par la machine, à partir de différents langages évolués ou pas.**

Ainsi, toutes les fiches sont identiques et possèdent le même format. Une fiche est aussi appelée un article. Dans un fichier clients, par exemple, chaque article correspond à un client particulier.

Un article est composé de données utiles au traitement et, dans l'exemple ci-dessus, une donnée peut être, le nom, l'adresse ou un numéro de téléphone.

L'ensemble des lignes d'un programme BASIC, par exemple, constitue en lui-même un fichier dont chaque article est caractérisé par un numéro d'ordre (le numéro de ligne) en tête et un **« retour chariot »** à la fin. Ce fichier est sujet à des modifications, mises à jour (M.A.J.), chaque fois que l'on ajoute une ligne intermédiaire (écrite à la suite des autres, mais munie d'un numéro de ligne permettant de l'insérer au bon endroit) ou que l'on efface une ligne.

Tous les articles d'un même fichier sont composés d'une suite de zones (les données ou rubriques). L'une de ces zones contient les informations utiles au classement ou tri du fichier.

Le plus souvent il s'agit d'un numéro d'article, lié parfois à l'information elle-même. Les comptes bancaires, par exemple, pourront être rangés par le numéro de compte lui-même. Rien n'empêche cependant de dissocier le numéro d'ordre de l'information. Ce numéro est appelé parfois **critère de tri.** Une organisation judicieuse et un critère de tri simple permettront par la suite de retrouver facilement un article.

## LA TAILLE

d'un enregistrement (dans le cas d'un support magnétique) ou article peut être fixe, auquel cas le fichier se prête à merveille à un traitement par blocs physiques.

Chaque enregistrement contiendra un élément servant à le délimiter et (ou) à définir l'emplacement de l'enregistrement suivant.

Dans l'exemple d'un fichier-programme BASIC, la longueur de chaque enregistrement ou élément de fichier est variable. Le délimiteur est le « Retour Chariot » (CR) et « Passage à la ligne suivante » (LF)

Lors du traitement d'un tel fichier, la recherche des éléments utiles dans chaque article passe par le décodage des caractères spéciaux délimitant chaque enregistrement. Ainsi, si l'on cherche le mot « FOR » dans un enregistrement, on cessera cette recherche, soit en le trouvant, soit en lisant l'ordre de fin d'enregistrement. La marche à suivre serait tout autre si l'on connaissait à l'avance la taille (constante ou indiquée en début de ligne) de l'enregistrement. Le test de fin de ligne n'aurait ainsi pas lieu à chaque caractère.

## LES SUPPORTS

Permettant le stockage non volatil des fichiers et leur disponibilité sont généralement magnétiques (bandes, disques ou disquettes, tambours, mémoires à bulles, etc.). On les trouve également sous forme de ruban ou de cartes perforées. Chaque support est caractérisé par sa capacité de stockage, sa vitesse de transfert son temps d'accès et son coût de stockage.

Chez les amateurs cela s'appelle mini-cassette, ou floppy disk.

## ORGANISATION DES FICHIERS

Dans le cas le plus général on trouve des fichiers organisés **séquentiellement** ou en **mode adresse.** Dans le premier cas, le séquencement pourra être physique ou logique.

Un exemple simple d'organisation en séquence physique est la bande magnétique. Les informations y sont mémorisées dans l'ordre ou elles se sont présentées lors de l'inscription. Pour retrouver un enregistrement on doit les lire tous. L'accès au $N^{\text{ième}}$ enregistrement physique impose la lecture des N-1 précédents. Les ordres physique et logique sont les mêmes.

Lors d'une organisation séquentielle logique on utilise un lien logique (zone pointeur contenant l'adresse de l'enregistrement suivant). Ce lien rend indépendantes les organisations logiques et physiques des fichiers.

La consultation et la mise à jour d'un tel fichier se feront à partir du premier enregistrement, en suivant la chaîne : si l'on cherche l'enregistrement « Paul », dans l'exemple de la **figure 1,** on partira du premier enregistrement qui est celui d'André :

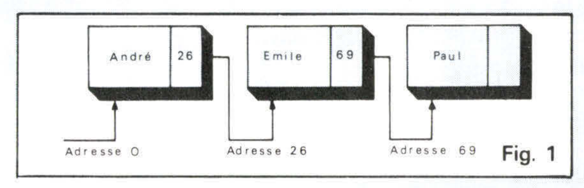


Fig. 1

Nous ne savons pas a priori que l'enregistrement « Paul » est à l'adresse 69, mais il suffit de suivre pour arriver à l'article voulu.

Etant donné l'accès-série, il faudra mettre en tête de liste l'enregistrement le plus souvent demandé.

Il y a d'autres modèles d'organisations logiques :

- **L'organisation circulaire :** le dernier enregistrement pointe vers le premier et il n'y a plus d'enregistrement « début », à chaque utilisation le pointeur restant positionné sur le dernier enregistrement utilisé.
- **L'organisation symétrique :** chaque enregistrement est muni d'une adresse pointant vers l'emplacement du suivant et d'une autre pointant vers l'enregistrement précédent comme le montre la **figure 2.**

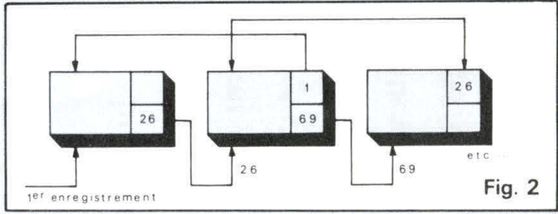


Fig. 2

Passons sur l'organisation en listes imbriquées, tri-directionnelles, multiples, inverses, etc., et retenons tout simplement qu'une organisation séquentielle logique permet d'éviter des manipulations physiques fastidieuses. Si l'on devait insérer lors d'une mise à jour quelques enregistrements, par exemple, ils pourront, dans le cas logique, être ajoutés à la suite des autres, sans avoir d'autres modifications à faire que celle de deux ou trois liens (numéros) de chaînage à l'endroit voulu de la chaîne.

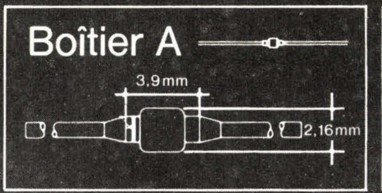
Ainsi, la longueur de l'adresse de chaînage est fixe. Les enregistrements à ajouter pourront avoir une taille quelconque, les modifications ne bouleverseront pas les anciens enregistrements.

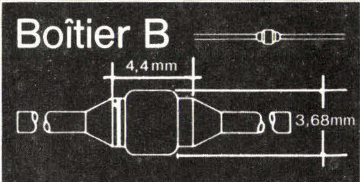
Un autre type d'organisation, similaire au mode d'adressage direct, demande de faire correspondre à chaque enregistrement, un emplacement physique. Une zone particulière de l'enregistrement, la **CLE,** permettra l'adressage (pouvant s'obtenir éventuellement par une formule de calcul) et donc l'emplacement physique de l'enregistrement. Ce cas est particulièrement intéressant si l'on dispose d'un support magnétique volumineux, car la taille fixe des enregistrements physiques fera intervenir de nombreux blancs, inutiles, portant le taux de remplissage à de faibles valeurs. ■

# BARRAGE AUX TRANSITOIRES




## Gain de place




## Economies

Entre 5 et 10 F. h.t., est-ce trop cher pour mieux protéger vos c.i. : µP, RAM, ROM, PROM..?

**Caractéristiques principales des suppresseurs de transitoires**

| Réf. | $V_c$ (Volts) | $I_{pp}$ (Amp.) | $P_c$ (1 m/s) (Watts) |
|---|---|---|---|
| **TVS - Boîtier B** | | | |
| TVS 505 | 9.3 | 53.7 | 500 |
| TVS 510 | 16.5 | 30.3 | |
| TVS 512 | 21.0 | 23.8 | |
| TVS 515 | 25.2 | 19.8 | |
| TVS 518 | 30.5 | 16.3 | |
| TVS 524 | 42.0 | 11.9 | |
| TVS 528 | 46.5 | 10.7 | |
| **UZS - Boîtier A** | | | |
| UZS306 | 8.7 | 17 | 150 |
| UZS312 | 16.8 | 8.9 | |
| UZS315 | 21.0 | 7.1 | |
| UZS318 | 25 | 5.9 | |
| UZS330 | 42 | 3.6 | |
| UZS333 | 46 | 3.2 | |
| UZS356 | 78 | 1.9 | |
| UZS426 | 355 | 0.42 | |
| UZS428 | 380 | 0.39 | |
| UZS440 | 545 | 0.28 | |

Disponibles sur stock

**Notes d'application U79 en préparation**

UNITRODE = LA PUISSANCE




**Tour EUROPA · Centre Commercial Belle-Épine - EUROPA 111**
**94532 RUNGIS Cedex - Tél. : 686.56.65 - Télex : 250801**

birgépub 644

Salon des Composants : Allée 14 Stand 141

# Les « microprocesseurs 16 bits »

**Quelques caractéristiques de microprocesseurs 16 bits**

| | DATA General $\mu$m 601 | DEC LSI 11 | Fair-child 9940 | General Instr CP 1600 | National PACE | Texas TMS 9900 | Intel 8086 | NEC $\mu$com 1600 | Motorola 68000 | Zilog Z8000 |
|---|---|---|---|---|---|---|---|---|---|---|
| Technologie | N.MOS | N.MOS | I3L | N.MOS | P.MOS | N.MOS | H.MOS | MOS | MOS | N.MOS |
| Capacité mémoire octets | 64 k | 64 k | 64 k | 64 k | 64 k | 64 k | 1 million | 1 million | 16 M | 8/48 M |
| Alimentation | + 14<br>+ 10<br>+ 5 | + 12<br>+ 5<br>- 12 | + 5 | + 12<br>+ 5<br>- 3 | + 5<br>- 12 | + 12<br>+ 5<br>- 5 | + 5 | + 5 | + 5 | + 5 |
| Boîtier | 40 | 40 | 40 | 40 | 40 | 64 | 40 | 64 | 64 | 40/48 |
| Remarques | | 4 boîtiers | | | | | bus multiplexé | | | bus multiplexé |

**La philosophie des constructeurs est très diversifiée dans le domaine des microprocesseurs 16 bits.**

**Intel a choisi la ligne de la continuité en évitant de troubler les utilisateurs des 8 bits, en les rassurant au maximum. Au contraire, Zilog et Motorola ont choisi la démarche inverse : leurs 16 bits constituent une rupture avec le 8 bits. Ils visent les mini-ordinateurs de haut de gamme du type PDP 11/45 avec toutes les implications que cela suppose au niveau software. Le véritable point commun de ces systèmes est le langage « officiel » choisi par chaque fabricant. Ils dérivent tous plus ou moins directement du Pascal.**

## A qui s'adresse ce type de composants ?

La première question qui se pose en observant leurs caractéristiques générales est :

A qui s'adresse ce type de composants ?

— à une clientèle équivalente à celle du 8 bits, c'est-à-dire très proche de l'électronique ?

— aux fabricants de systèmes informatiques qui remplaceront leurs unités centrales par ces microprocesseurs ?

Le marché visé est celui de la mini-informatique : le coût très faible (comparativement au logiciel) du matériel permettra sans doute à des sociétés de service qui développaient uniquement du logiciel sur des mini-ordinateurs de s'orienter vers une machine personnalisée (matériel et logiciel). Par contre, il existe des applications qui, jusqu'ici, relevaient de grosses machines (reconnaissances de formes, par exemple) qui pourront faire l'objet de réalisations à coût faible et productibles en série.

## 8 bits ou 16 bits ?

Autre question importante : y a-t-il concurrence entre les 8 bits et les 16 bits ?

Beaucoup « d'applications 8 bits » sont du domaine 8 bits. Toutes les applications, traitement et transmission de caractères, etc., relèvent intrinsèquement de ce domaine et un microprocesseur du type 8085 ou Z 80 nettement plus rapide serait le bienvenu et peut-être plus approprié qu'un 16 bits.

Le marché supposé des 16 bits tournerait entre 10 et 20 % du marché des 8 bits, au moins au niveau des estimations des fabricants. Ces chiffres paraissent très théoriques et semblent relever plutôt du domaine du pari que de considérations objectives.

## Evolution technologique

L'annonce de la complexité des microprocesseurs 68000 (70000 transistors) ou 8086 (29000 transistors) amène la question suivante : à la vitesse d'évolution actuelle des technologies, quelle sera la durée de vie industrielle de ces composants ? La question est importante, non pas au niveau hardware, mais surtout au niveau logiciel. Le développement logiciel est très coûteux, donc très long à amortir et, avec le développement software, il y a intérêt à choisir le composant qui sera opérationnel dans 5 ou 10

Il est remarquable de constater l'unanimité de pensée concernant le type de langage implanté dans les systèmes à microprocesseur 16 bits.



ans. L'évolution des mémoires prévoit le Mégabit aux alentours de 1985. Le champ d'adressage du microprocesseur devra être au moins égal à ce chiffre. C'est le cas des machines de nouvelle génération (8086, Z 8000, 68 000). Le problème de la technologie est lié aussi à la faisabilité et au rendement de fabrication des composants. La guerre des fabricants impose des annonces de sortie sur le marché avec des dates plus ou moins glissantes.

La technologie H.MOS Intel a fait ses preuves, puisque le 8086 existe, mais le Z 8000 sera-t-il industrialisé à court terme ? Pour Motorola, l'étape doit être encore plus marquée (deux fois plus de transistors que pour le 8086).

A long terme, une autre question s'impose : quelle sera l'étape après le 16 bits : le 32 bits ? Mais là le marché se rétrécit sérieusement. Le marché des 16 bits est estimé à 10-20 % des 8 bits. Pour le 32 bits cela tombe à quelques pour cent. Le jeu en vaut-il la chandelle ? Les estimations du coût en recherche et développement des 16 bits tournent entre 10 et 15 millions de dollars. Il est évident que le marché ne doit pas se réduire à quelques milliers de pièces.

Les successeurs seront-ils les machines orientées langage (voir problème de logiciel) qui sont dès maintenant possibles, avec des avantages intrinsèques évidents.

## Logiciel

Il est à peu près évident que le logiciel qui sera développé sur les systèmes 16 bits sera différent du software pour les 8 bits.

Les 8 bits ont été et sont utilisés en grande partie en séquenceur programmable (gestion d'entrées-sorties, etc.) avec quelques fonctions de calcul, mais très peu, sauf dans le cas de construction de micro-ordinateur à vocation mini-ordinateur bas de gamme. Le langage le plus adéquat qui « colle » bien au fonctionnement très électronique des systèmes est l'assembleur, avec tous les avantages et les inconvénients de la programmation assembleur. Pour des programmes compris entre 0,5 k et 4 k-octets, la nécessité d'un langage évolué n'est pas évidente. Au contraire, le taux d'expansion de ces langages implique un n-tuplement de la mémoire programme (2 à 10 suivant le langage) d'où un coût élevé et une diminution des performances en vitesse d'exécution.

Il existe aussi de petites applications 16 bits où le microprocesseur 16 bits a un rôle prépondérant de séquencement, l'utilisation du 16 bits n'étant imposée que par la puissance supérieure du jeu d'instructions (cas de la multiplication division). La méthodologie 8 bits reste alors transportable et avantageuse.

Par contre, dans les domaines où le microprocesseur 16 bits aura fonction de remplacer un mini-ordinateur, il est fondamental de changer de méthode sous peine de graves déboires.

Le passage aux langages évolués devient une nécessité, surtout pour des problèmes de maintenance et de coût de réalisation.

Existe-t-il un langage commun à tous les microprocesseurs 16 bits ? Il est remarquable de constater l'unanimité de pensée vers un langage ou du moins un type de langage qui est ou sera implanté. Ces langages dérivent tous du **Pascal ;** langage créé par Niklaus Wirth en 1970. Ses caractéristiques sont tirées des constatations suivantes : les langages de haut niveau Fortran, Basic, s'ils sont très facilement utilisables posent des problèmes au niveau maintenance, relisibilité, reprises des programmes. Ils permettent comme dit Webster dans son « Introduction to Pascal » d'acquérir un « disorganized unstructured » style de programmation. Le Pascal, par sa syntaxe, son architecture impose une structuration de la programmation et en sous-tend une méthodologie plus rationnelle.

Texas et Motorola ont choisi officiellement le Pascal comme outil de programmation. Le PLZ de Zilog est dérivé du Pascal, en conservant la même philosophie. Cette universalité du choix peut s'expliquer pour plusieurs raisons :

- pour beaucoup d'applications qui seront faites par des « non-informaticiens » de formation, il est important de choisir un langage qui évite un certain nombre de défauts inhérents au langage lui-même ;
- écriture facilitée des compilateurs (hypothèse de définition du Pascal par Wirth).

Si on peut espérer une transportabilité des logiciels au niveau des langages (en restant très prudent sur cette notion), il reste dans l'hypothèse des machines langages des années 80 que le Pascal se prête bien à ce genre d'évolution. Dès aujourd'hui, Western Digital fabrique un microprocesseur qui décode directement le P code issu du Pascal (codage généralement intermédiaire obtenu à partir du compilateur Pascal). A ce niveau, le compilateur est universel, car il traduit un langage source en un code ne dépendant pas de la machine. La spécificité de la machine intervient uniquement, sauf dans le cas de Western Digital, au passage code P-code machine.

## Compatibilité ?

La conversion des softwares 8 bits en solftware 16 bits est peu vraisemblable pour les raisons suivantes :

- applications différentes,
- les applications assembleurs ne doivent pas être transcrites au niveau de l'instruction, car il est évident que l'on ne disposera pas alors de la surpuissance au niveau composant et même pour des microprocesseurs à parenté très proche et voulue du type 8080-8086 le parallélisme est utopique. Il n'y a qu'à examiner les tables de correspondances d'Intel.

**Exemple :**

| Instruction 8080 | Instruction 8086 |
|---|---|
| DAD reg addition sur double registre | LAHF<br>ADD BX, reg<br>RCR SI<br>SAHF<br>RCL SI |

A part l'aspect langage, l'utilisation des microprocesseurs 16 bits

## Le Séminaire E.S.I.E.E.

Maintenant que les microprocesseurs 16 bits existent, il ne s'agit pas de s'attarder sur le fait de savoir qui les fabrique.

Il est, par contre, urgent d'entreprendre une vaste campagne de sensibilisation auprès de nos industriels français de façon à ce qu'ils réalisent toute la richesse potentielle que contiennent ces composants en leur offrant la possibilité d'apporter dans l'utilisation de ceux-ci une forte valeur ajoutée, fruit d'un savoir faire propre à placer notre pays dans le peloton de tête de l'économie mondiale.

C'est la raison pour laquelle l'E.S.I.E.E. * organisera, conformément à sa vocation de formation et d'information, le 29 mars, un séminaire d'une journée sur le thème « les microprocesseurs 16 bits pour qui, pour quoi ? ».

Cette journée comprendra deux volets :

● exposés des sociétés : Digital, Intel, Motorola, Texas, Zilog ;

● une table ronde sur le thème « l'industrie française et les 16 bits », couplée à une exposition qui présentera des produits réalisés avec des microprocesseurs 16 bits.

A cette occasion, Micro-Systèmes sera parmi les exposants.

impliquera dans la majorité des cas, un système d'exploitation. Au niveau des 8 bits, cela n'est pas évident. La petitesse de la plupart des applications, leur diversité, leur spécificité, imposait une programmation « sur mesure » pour optimaliser la taille mémoire. Au contraire, pour les 16 bits, la nécessité de standardisation du logiciel exigera des O.S plus ou moins sophistiqués, adaptés à l'application. Les microprocesseurs émulants des machines existantes ont un avantage certain sur les machines nues du type 8086, Z 8000 et 68000, même si leur architecture hardware est ancienne, la capacité d'adressage faible, la récupération de centaines d'heures homme de logiciel ne peut échapper à tout concepteur. Par exemple, pour DEC, l'argument massue consiste à dire : en choisissant le LSI 11, vous disposez non seulement du jeu d'instructions des PDP 11 (qui, entre parenthèses, sert de référence ou de points de comparaison aux autres micros), mais vous héritez (en partie) de milliers de mois homme logiciel.

* *E.S.I.E.E. (Ecole Supérieure d'Ingénieur d'Electronique et d'Electricité), 81, rue Falguière, 75015 Paris. Tél. : 320.12.15 poste 342.*

Cet argument est surtout à prendre en compte pour des applications à court terme pour lesquelles il n'est pas question de développer tout un software de base important pour le 8086 ou les 68000 et Z 8000 à venir. Le problème sera différent dans deux ou trois ans car le déséquilibre sera moins prononcé alors.

## Choix

A ce niveau, pour l'utilisateur, quel sera le critère de choix entre mini et micro ? Les caractéristiques hardwares sont voisines (à comparer les benchmarks Z 8000, 68000, PDP 11/45). Il apparaît que le mini-ordinateur est une machine complète hard-software avec une périphérie adéquate alors que les microprocesseurs actuels 16 bits sont plus dépouillés. Dans les années à venir, cette différence s'atténuera avec, en contre partie, une montée en puissance des minis (d'ailleurs les termes de mini, micro, etc., n'ont plus aucun sens) et un développement de la périphérie spécifique pour ces composants nouveaux. Les fabricants de minis auront à répondre au dilemme suivant :

● suivre la voie de DEC ou Data General en fabriquant ou se faisant fabriquer son microprocesseur à partir de son jeu d'instructions ;

● intégrer les nouveaux composants : cela sera difficile, car il n'y aura aucune compatibilité avec les machines précédentes du constructeur ;

● faire le choix des microprocesseurs en tranches type AMD 2900 (ou suivants) qui sauvegarderont « l'originalité » du jeu d'instructions et maintiendront une compatibilité avec les gammes déjà existantes.

A priori ces solutions ne seraient pas plus performantes à tâches égales avec les microprocesseurs intégrés déjà existants et surtout seront plus chères à la réalisation, mais sera-ce là le prix de l'indépendance vis-à-vis des fabricants américains ?

## Composants ou kit ?

Pour un concepteur, le premier problème à envisager sera :
— acheter le composant
— acheter le kit constructeur.

Le choix du composant est rentable pour une série (au-delà de 100 ?). Il permet d'autre part d'être indépendant du constructeur de la plaque, d'où la possibilité de secondes sources pour l'approvisionnement des composants. Par contre, ce choix impose un service de maintenance efficace.

Le choix du kit constructeur favorise une mise en œuvre plus rapide et permet de bénéficier d'une part des plaques de la famille et d'autre part, des softs systèmes (écrits ou à venir) compatibles avec ces kits.

## Investissements matériels

Le passage aux 16 bits nécessitera des outils de développement plus puissants, qui seront en fait équivalents à des minis actuels et disposant de systèmes d'exploitation et d'outils de mise au point plus performants que les outils utilisés pour les 8 bits.

Le coût de ces outils de développement conduira peut-être à une nouvelle conception du mode de travail : configuration « centre de calcul » avec périphérie importante et postes de travail sous « time sharing ». Ceci est déjà applicable pour le LSI 11. Nous avons d'autre part utilisé cette méthode pour notre enseignement microprocesseur 8 bits à l'E.S.I.E.E. 8080, 8085, Z 80 (cross assembleur et simulateur implantés sur PDP 11/70 avec 24 postes de travail).

Par ces remarques préliminaires, nous avons voulu soulever quelques problèmes que tout responsable de projet doit se poser pour que le passage aux 16 bits se fasse sans trop de « douleurs ». ■

**J. CALLOT**
**M. CODOU**

# Micro Electronique - Micro Informatique

### INFORMATIQUE
***D.J. DAVID***

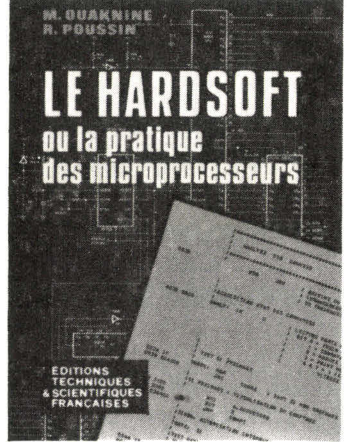
Cours d'initiation à l'informatique (ENS). Langages de programmation : Fortran, APL. Fonctionnement interne des ordinateurs. L'esprit informatique, modèles schématiques des applications, cartes-contrôle : IBM, CDC, UNIVAC, CII et Philips, 336 pages.

***NIVEAU 3*** ***PRIX : 65 F***

### MICRO-INFORMATIQUE MICRO-ELECTRONIQUE DICTIONNAIRE
***LILEN et MORVAN (I.C.S.)***

Un millier de mots, sigles et expressions. Définitions françaises et leur traduction (français-anglais). Lexique anglais-français. 370 pages.

***NIVEAU 2*** ***PRIX : 92 F***




### TECHNIQUES D'INTERFACE AUX MICROPROCESSEURS
***LESEA et ZAKS (SYBEX)***

Comment connecter un système à microprocesseur aux périphériques, depuis l'unité centrale jusqu'au clavier, télétype, disque souple, écran de visualisation, et interfaces analogiques. Techniques de test. 416 pages.

***NIVEAU 2*** ***PRIX : 95 F***

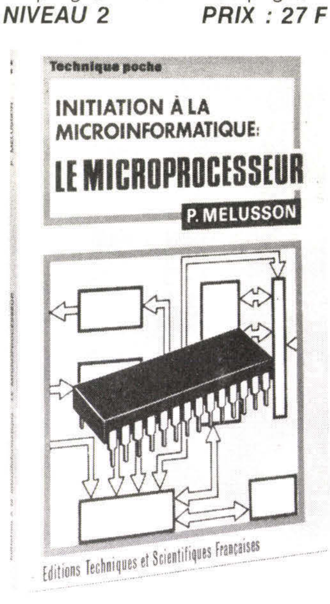



**ÉDITIONS TECHNIQUES ET SCIENTIFIQUES FRANÇAISES**
**Ventes libraires. 2 à 12, rue de Bellevue**
**75940 Paris Cedex 19**

### TECHNIQUE POCHE N° 4
### INITIATION A LA MICROINFORMATIQUE LE MICROPROCESSEUR
***P. MELUSSON***

Qu'est-ce qu'un ordinateur. Langages. Calcul binaire. Codages. Fonctions logiques. Technologie et organisation des microprocesseurs. Les mémoires. Circuits et systèmes d'interface. La programmation. 136 pages.

***NIVEAU 2*** ***PRIX : 27 F***




### LE HARDSOFT ou la PRATIQUE des MICROPROCESSEURS
***M. OUAKNINE et R. POUSSIN***

Principes généraux. Fonctionnement et jeu d'instruction d'un système construit autour d'un microprocesseur 8080A. Trois applications réelles avec schémas et programmes. Fonctionnement des dernières nouveautés 8048-Z80 - 8086. 254 pages.

***NIVEAU 3*** ***PRIX : 65 F***

### LEXIQUE MICROPROCESSEURS
***(SYBEX)***

Dictionnaire anglais-français. 1 000 termes et abréviations. Définitions des composants par numéros, des signaux pour les bus S 100, RS 232C, IEEE 488. Adresses des fabricants et distributeurs. Table de conversion. Format Poche. 120 pages.

***NIVEAU 2*** ***PRIX : 20 F***

### INTRODUCTION AUX MICROORDINATEURS INDIVIDUELS ET PROFESSIONNELS
***R. ZAKS (SYBEX)***

Ce livre vous permettra d'évaluer si vous devez utiliser l'un des nouveaux microordinateurs.
Comment choisir son système.
Définitions, pièges à éviter, programmation. Quel Basic ?
— Applications professionnelles et commerciales
— Choix des périphériques.

***NIVEAU 1*** ***PRIX : 52 F***

### LOGIQUE INFORMATIQUE
***M. FERRETTI***

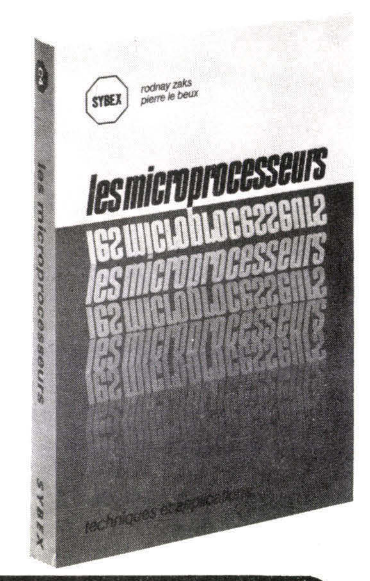
— Qu'est-ce qu'un ordinateur.
— Cours et exercices sur la théorie des ensembles. Lois de composition. Relations binaires. Multiplication, puissance des nombres relatifs.
— L'Algèbre de Boole. Logique de commutation. Représentation et minimalisation des fonctions booléennes. 160 pages.

***NIVEAU 3*** ***PRIX : 25 F***

### LES MICROPROCESSEURS
***ZAKS et LE BEUX (SYBEX)***

Ouvrage de base conçu pour la formation. Concepts et techniques. Principes de bases jusqu'à la programmation. Techniques « standards ». L'interconnexion d'un système « standard ». Les problèmes liés au développement d'un système. 320 pages.

***NIVEAU 2*** ***PRIX : 95 F***




### ELEMENTS ESSENTIELS DE L'ELECTRONIQUE ET DES CALCULS DIGITAUX
***D. ULRICH***

Logique électronique. Logique informatique. Calculateurs à circuits logiques. Réalisation des calculateurs. Le transistor en commutation. Multivibrateurs. Montages logiques de base. Fonctions logiques. Algèbre de Boole. Calculs binaires. 304 pages.

***NIVEAU INGENIEUR*** ***PRIX : 86 F***



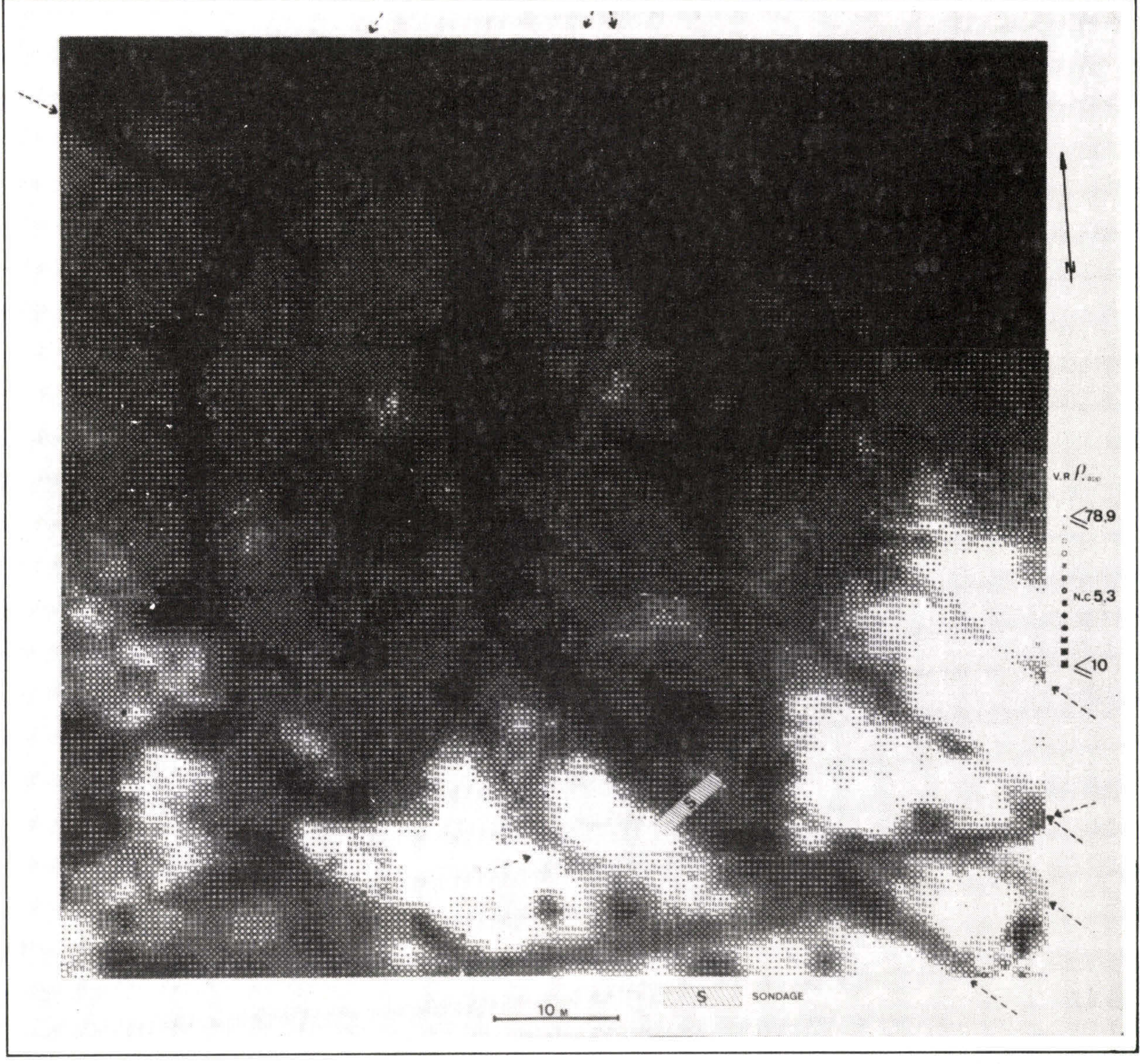



*En vente chez votre*
**LIBRAIRE HABITUEL**
**ou à la LIBRAIRIE PARISIENNE de la RADIO**
**43, rue de Dunkerque, 75480 Paris Cedex 10**

**AUCUN ENVOI contre remboursement. Port : Jusqu'à 25 F : taxe fixe 3,50 F. De 25 F à 100 F : 15 % de la commande (+ 3,50 F Rdé). Au-dessus de 100 F : taxe fixe 18,50 F.**

***NIVEAU 1 : Initiation***
***NIVEAU 4 : Ingénieur***
***Tarif : Décembre 1978***

# Le micro-ordinateur et la recherche archéologique




*Exemple de traitement de l'information saisie en prospection géophysique. Résultats d'une prospection électrique sur un site néolithique en Touraine. Cette cartographie met en évidence les variations de la résistivité électrique apparente du sous-sol sur une partie d'un camp préhistorique. 6000 mesures au sol couvrant 6 000 m² ont été nécessaires pour cette investigation. Ce document fait apparaître une importante organisation du sous-sol. En noir, on peut distinguer les structures qui constituent ce campement enfoui à 0,50 m sous la surface du sol (fossés, fosses comblées et zones d'habitat).*

Liée au progrès de la micro-informatique et au champ d'application extrêmement vaste qu'offrent désormais les micro-ordinateurs, l'informatique devrait, en archéologie, et plus particulièrement dans le domaine de la prospection, connaître des développements importants. Découvrir des sites archéologiques enfouis, préciser leurs plans et leurs natures, tels sont les principaux objectifs de cette discipline (recherche aérienne, prospections géophysiques, répartition de surface des témoins archéologiques).

Si l'informatique n'est pas absolument nécessaire pour deux de ces méthodes d'investigation du sous-sol, ses applications sont nombreuses et indispensables dans le domaine de la prospection géophysique.

En effet, l'intervention de ces techniques (prospections magnétiques, électriques et électro-magnétiques) nécessite dans la plupart des cas la saisie et le traitement de données dont le nombre peut varier d'une dizaine à plusieurs centaines de milliers. C'est ainsi que depuis 1972, l'exécution par notre laboratoire d'une trentaine de missions en France et à l'étranger (Italie, Maroc, Espagne, Afghanistan) a nécessité l'acquisition et la restitution cartographique d'un peu plus de 400 000 données numériques. Sollicités de plus en plus fréquemment à mettre en application d'une manière extensive ces procédés de détection, nous avons entrepris récemment de développer le système de traitement de l'information que nous utilisions jusqu'ici, lequel n'était désormais plus adapté aux exigences de notre discipline.

Mis au point et fonctionnant avec satisfaction pour les besoins correspondant à une phase de recherche expérimentale, ce système de stockage et de représentation graphique était constitué d'une calculatrice Wang 720 C, programmable en langage machine, connectée à une impri-

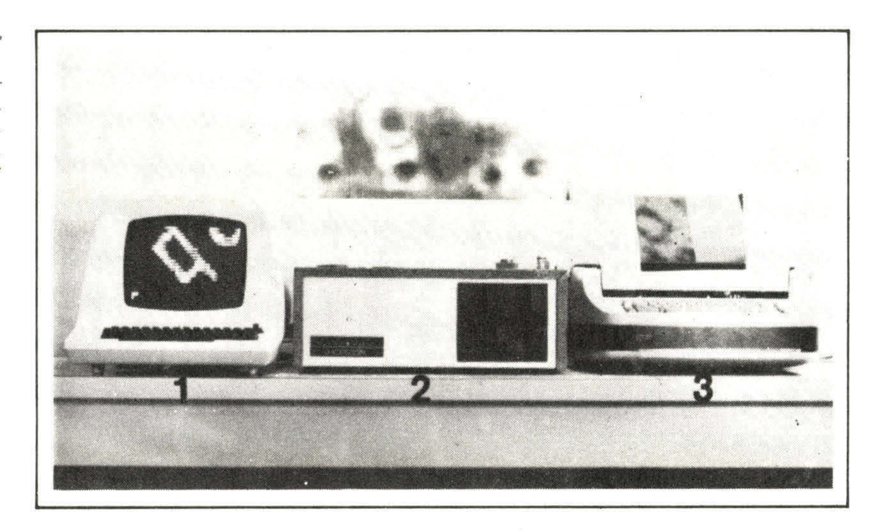



*Photo 1. - Ensemble de l'unité de traitement : 1. console Lear-Siegler ADM3 - 2. micro-ordinateur North-Star-Horizon - 3. imprimante Wang-IBM Selectric.*

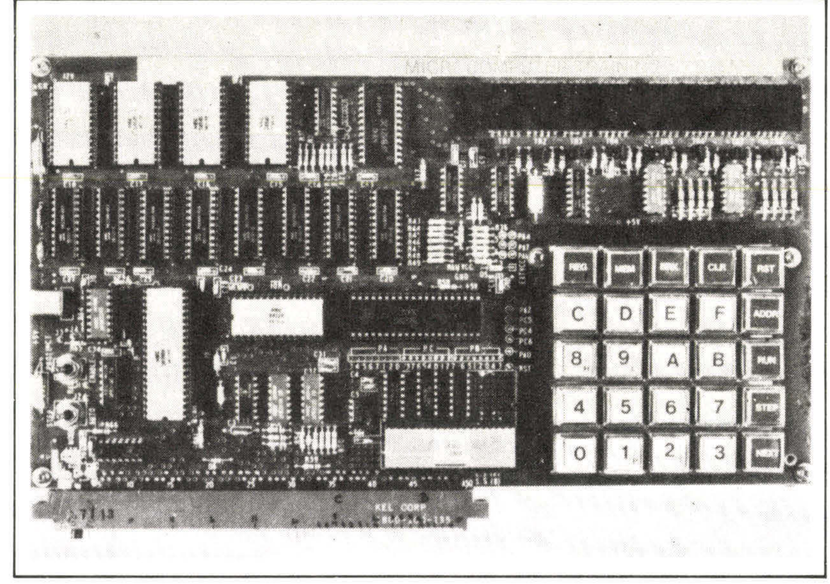



*Photo 2. - Unité Centrale du poste de saisie de données NEC TK-80.*

mante à boule IBM Wang 701. Transitant par la mémoire centrale en tores de ferrite de 2 K instructions, les données à traiter, introduites par le clavier d'une manière séquentielle, étaient stockées en mémoire de masse par l'intermédiaire d'un enregistreur/lecteur magnétique.

Actuellement encore utilisée sur notre nouveau système, la cartographie des variations de ces grandeurs physiques mesurées au sol est basée sur l'utilisation des propriétés de la représentation symbolique. Cette méthode consiste à créer une image mono ou multicolore dont les variations de tons permettent de visualiser directement l'amplitude des phénomènes à mettre en évidence.

L'utilisation et la combinaison de caractères de l'alphabet grec et de symboles mathématiques permettent d'obtenir des nuances (13, 26, 39) en représentant chaque point du réseau de mesure par un caractère dont « l'intensité » est proportionnelle à l'amplitude. L'incrément (amplitude représentée par un caractère ou niveau) peut être linéaire ou dépendre d'une fonction particulière. Cependant, dans le cas de résultats obtenus sur des sites archéologiques complexes, ce seul moyen de traitement de l'information ne peut permettre d'obtenir des documents où il soit aisé de discriminer le « bruit » des structures recherchées. Ce bruit, d'origines diverses (électronique, effets géologiques, agraires, topographiques, de destruction...) qui se superpose au signal émis ou induit par les formations archéologiques ordonnées, peut être estompé ou parfois éliminé en utilisant les propriétés de certains algorithmes mathématiques ou techniques de filtrage optique.

Ceci est donc la deuxième raison pour laquelle l'emploi de l'ordinateur est indispensable dans cette discipline.

## Mise en application d'un micro-système

La présente configuration de ce système multilangage, standardisé BUS-S-100 est la suivante : une unité centrale Z80, 32 K octets de mémoire centrale statique, un interface entrée/sortie 3P+S, un interface analogique digital, 2 stations mini-floppy disk de 90 K0, un terminal écran/clavier Lear-Siegler ADM3, une imprimante IBM Wang.

Mis au point au laboratoire, le logiciel actuellement opérationnel comprenant 40 programmes écrits en **Basic** et **langage machine,** permet de résoudre les phases essentielles de stockage de l'information et d'effectuer des traitements numériques spécifiques (interpolation linéaire, statistiques, filtrage par corrélation, etc.).

Parallèlement à la réalisation de cette chaîne de traitement autonome, nous étudions la possibilité d'une saisie et d'un procédé de stockage automatique des données de la prospection.

La version que nous avons choisie de réaliser comprend une carte (NEC KIT TK.80) pilotée par un micro-processeur Intel 8080 et interfacé à un enregistreur magnétique rapide (Digital Group Phydeck). Cette configuration devra permettre **sur le terrain même** de gérer automatiquement la saisie et le stockage des informations issues de capteurs géophysiques et, dans certains cas, d'effectuer un prétraitement numérique. Les conséquences et les avantages essentiels de l'utilisation de la micro-informa-

tique dans notre discipline sont multiples :

- Concernant la phase d'acquisition de l'information, le micro-système de saisie permettra de s'affranchir de toute manipulation de données. Cette automatisation devra d'une part accroître la précision des mesures physiques et, d'autre part, diminuer d'un rapport 2 le temps habituellement nécessaire à l'exécution et au traitement d'une prospection.
- Si l'informatique est utilisée depuis quelques années en archéologie, ses applications n'en sont pas moins restées limitées, en raison principalement des problèmes que posent l'accès aux moyens disponibles et du coût d'exploitation bien souvent trop élevé.

L'acquisition de ce micro-ordinateur a permis à notre laboratoire d'être autonome et polyvalent. Cette notion d'autonomie est un fait très important car elle permet désormais de s'affranchir des aléas de la sous-traitance et de pouvoir ainsi personnaliser chaque phase nécessaire à ce type de recherche. Cette considération est d'autant plus importante que l'étude d'un site est dans la plupart des cas une recherche spécifique.

En effet, et indépendamment de sa nature, chaque site archéologique est unique, tant par sa situation géographique que par ce qu'il nous en reste. Ainsi, les paramètres à prendre en considération à chaque traitement étant innombrables et différents, l'élaboration des résultats ne peut s'effectuer que par approches successives avec un temps d'utilisation machine important. Dans ces conditions, l'autonomie informatique est fondamentale car la majorité des centres de calcul qui nous sont accessibles ne peuvent répondre, dans les meilleurs délais, à nos besoins et dans le cas de ceux qui le pourraient, le prix d'exécution dépasse nos possibilités financières. La configuration de ce système n'est pas figée, sa conception BUS-S-100 permet une extension directe à partir des très nombreuses cartes accessoires de périphérie qui existent actuellement sur le marché et à des prix raisonnables.

Fonctionnant avec entière satisfaction depuis cinq mois, cet équipement, dont le coût est d'environ 30 000 F, nous a permis de progresser très nettement dans notre recherche. ■

**J. HENRY (1)**
**P. GOURMEL**
**A. KERMORVANT**
**J. MASSON**

*(1) J. Henry est directeur du Laboratoire de Physique de la Faculté des Sciences de Tours. P. Gourmel, A. Kermorvant et J. Masson sont chercheurs auprès de ce laboratoire.*

# PROTEUS III A




**PROTEUS III UNITÉ CENTRALE**
16 K-Ram. (Réf. Proteus III A)

**Prix .................. 7 500 F H.T.**

32 K-Ram. (Réf. Proteus III B)
**Prix .................. 9 148 F H.T.**
1 K-Mémoire de page écran.
8 K-Rom (Basic).

Unité centrale 6800 travaillant à 0,894 MHz. Clavier 53 touches capacitives. 128 caractères dont ASCII majuscules + caractères graphiques.
Sortie UHF et vidéo.
Interface imprimante série : 20 mA, TTY, RS 232, gérée par le Basic.
Basic type étendu, capable de gérer 3 périphériques simultanément (clavier-écran vidéo, imprimante-TTY, magnéto K7), plus éventuellement 1 à 3 floppy disk.
Magnétophone à K7 travaillant à 300 bauds, utilisé comme mémoire de masse.

**PROTEUS MONITOR**
Monitor vidéo de 30 cm.
**Prix ................. 1 480 F H.T.**

**PROTEUS SPRINT**
Imprimante sur papier normal (non métallisé), travaille sur 80 colonnes à 1200 bauds.
**Prix .................. 8 707 F H.T.**

**PROTEUS FLOPPY**
Mini-Floppy travaillant en double densité, double face. 2,6 mégabits par disquette.
Vitesse de transfert de 250 K-bauds.
Temps d'accès moyen à un fichier : 500 ms.
**Prix .................. 6 650 F H.T.**



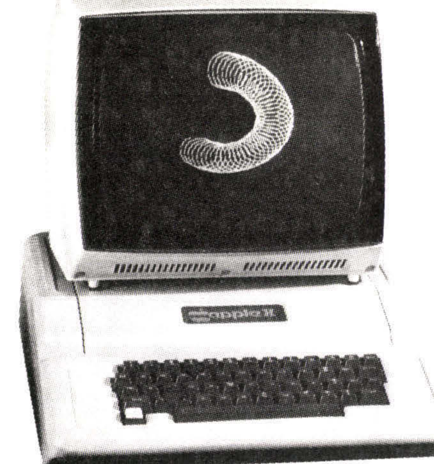

# MICRO INFORMATIQUE

**143, AVENUE FÉLIX-FAURE, PARIS XVe**
**Tél. : 554.22.22 - 554.83.81**

*DÉMONSTRATION LE SAMEDI APRÈS-MIDI OU SUR RENDEZ-VOUS*

**ATTENTION.** — Les prix cités étant hors taxe, il y a lieu de les majorer de 17,60 %.

# APPLE II

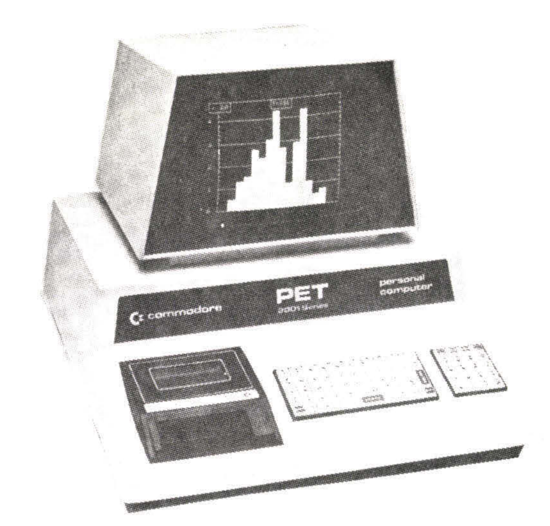



## MICRO INFORMATIQUE
**I. — MICRO-ORDINATEURS et FLOPPY DISKS APPLE II**

**APPLE II** est un micro-ordinateur complet, assemblé et testé. Il est livré avec câbles de raccordement, manuel d'utilisation, cassettes de démonstration, ainsi que deux manettes pour tracés de diagrammes.
**Configuration :**
16 K .................... **8 300 F H.T.**
32 K .................... **10 000 F H.T.**
48 K .................... **12 000 H.T.**
Vous pouvez nous consulter pour des configurations différentes.
**MINI FLOPPY « DISK II »**
D'une capacité de 116 octets formatés, d'un accès direct rapide, l'ensemble comprenant le Driver, le Contrôleur et le Disc Operating System (D.O.S.), est livré avec manuel et deux disquettes ...................... **4 950 F H.T.**

**II. — INTERFACES**
- Interface de codage couleur RVB (nécessite une prise RVB sur le téléviseur) .................. **780 F H.T.**
- Interface de codage couleur SECAM. **Prix** .................. **1 280 F H.T.**
- Modulateur noir et blanc seul .................... **280 F H.T.**
- Interface de sortie parallèle pour imprimante ou applications diverses .................... **1 280 F H.T.**
- Interface de série en mode V 24, RS 232 C (vitesse commutable de 75 à 19 200 bauds permettant le raccordement à tous les périphériques ou systèmes au standard de communication RS 232 C) ......... **1 280 F H.T.**

# P.E.T. DE COMMODORE




## UN SEUL COFFRET
**intégrant l'écran, le clavier, le magnétophone.**
**le P.E.T. de Commodore**

Complet, compact, le **P.E.T.** est particulièrement adapté à **l'enseignement,** à **l'industrie** et aux **laboratoires d'instrumentation** (bus IEEE 488). Basic puissant et rapide pour le **calcul.** Son prix le rend accessible aux utilisateurs individuels.
- Ecran incorporé à affichage très fin.
- Lecteur-enregistreur de cassettes standard incorporé.
- Clavier 73 touches avec symboles graphiques.
- Basic étendu résident avec grandes facilités d'édition.
- Interface IEEE 488.
- Connecteur d'accès à un port de 8 lignes d'entrée/sortie bidirectionnelles compatibles TTL, programmables.
- Connecteur d'accès à tous les bus du microprocesseur.

**Le système complet**
**6450 F H.T.**
**avec 16 K octets de ROM, 9 K octets de RAM dont 7 K disponibles pour l'utilisateur**
Manuel d'utilisation en français.

# Jeu de la Bourse




*La Bourse à Paris (Document Compagnie des Agents de Change, Paris, Photo J. LEHODEY).*

Avez-vous déjà participé à une séance de bourse, telle qu'elle nous est présentée par les caméras de télévision, avec son agitation et son ambiance de fièvre ? Si non, aimeriez-vous retrouver cette atmosphère avec votre ordinateur ? C'est ce que nous vous proposons dans ce numéro, avec une simulation très réaliste des péripéties et des aléas des revers de fortune.

Le principe en est simple : vous disposez de 2 000 F au début du jeu, et vous avez l'intention de faire fructifier votre avoir, afin d'arriver à la somme de 5 000 F. Plusieurs personnes peuvent jouer simultanement, et dans ce cas, il faut atteindre ce but le premier. Pour cela, chacun peut acheter ou vendre des actions (au moment opportun), afin d'augmenter son portefeuille.

L'évolution des cours est liée à différents facteurs :

- A chaque tour de jeu, une variation aléatoire fait monter ou baisser légèrement la valeur de chacune des actions.
- Chaque vente par un joueur fait baisser le cours de l'action concernée.
- De même chaque achat entraîne un renchérissement.
- Parfois un effondrement général des cours modifie profondément la situation financière respective des joueurs.
- La dévaluation de la monnaie est assimilée à une perte de 20 % sur les sommes dont on dispose en liquide.

# LISTING DU PROGRAMME

```
100 REM JEU DE LA BOURSE
110 GOSUB 5000 : REM EFFACEMENT DE L'ÉCRAN
120 PRINT TAB (10); « JEU DE BOURSE »
130 PRINT
140 PRINT « VOUS DISPOSEZ AU DÉPART DE 2 000 FRANCS »
150 PRINT « LE BUT DU JEU EST D'ARRIVER à 5 000 F LE PREMIER »
160 PRINT « POUR CELA VOUS POUVEZ ACHETER OU VENDRE »
170 PRINT « DES ACTIONS APPELÉES A, B, C, ETC »
180 PRINT
190 PRINT TAB (10); « BONNE CHANCE ! »
200 PRINT
210 INPUT « COMBIEN Y A-T-IL DE JOUEURS » ; J9
220 PRINT « MERCI »
230 IF J9 = 0 THEN 2 210
240 N9 = 5 : REM NOMBRE D'ACTIONS DIFFÉRENTES
250 P9 = 1/15 : REM PROBABILITÉ DE CRASH
260 P8 = 1/20 : REM PROBABILITÉ DE DÉVALUATION
270 DIM N $ (J9) : REM NOMS DES JOUEURS
280 DIM A (J9, N9) : REM NOMBRE D'ACTIONS ACHETÉES
290 DIM Q (J9, N9) : REM PRIX D'ACHAT DE CELLES-CI
300 DIM S (J9) : REM SOLDE DE CHACUN DES JOUEURS
310 DIM V (2, N9) : REM VALEURS DES ACTIONS
320 PRINT
330 FOR I = 1 TO J9
340 PRINT « NOM DU JOUEUR » ; I ;
350 INPUT N $ (I)
360 NEXT I
370 PRINT
390 REM BOUCLE DES PARTIES
400 FOR I = 1 TO J9
410 FOR J = 1 TO N9
420 A (I, J) = 0
430 NEXT J
440 S (I) = 2 000
450 NEXT I
460 REM INITIALISATION DES VALEURS
470 FOR I = 1 TO N9
480 V (1,I) = INT (RND * 20 + 20)
490 V (2, I) = 0
500 NEXT I
510 C = 0
580 REM BOUCLE DES COUPS JOUÉS
590 REM ON COMMENCE PAR AFFICHER LES COURS
600 GOSUB 5000
610 IF C = 1 THEN PRINT « CRASH GENERAL »
620 PRINT TAB (10); « COURS ACTUELS »
630 PRINT TAB (12); « VALEUR » ;
640 PRINT TAB (25); « VAR. »
650 FOR I = 1 TO N9
660 PRINT CHR $ (I + 64);
670 PRINT TAB (15); V (1, I);
680 PRINT TAB (24); « » ;
690 IF V (2, I) = 0 THEN PRINT « » ;
700 IF V (2, I) > 0 THEN PRINT « + » ;
710 PRINT V (2, I)
720 NEXT I
730 PRINT
740 REM TIRONS AU SORT QUI COMMENCE
750 A = INT (RND * J9 + 1)
760 A1 = A
770 PRINT N $ (A1);
780 PRINT TAB (13) ; « » ;
790 INPUT R $
800 IF R $ = « » THEN 1 340
810 IF ASC (R $) > 64 THEN 840
820 R $ = MID $ (R $, 2)
830 GOTO 800
840 IF LEFT $ (R $, 1) = « A » THEN C = 1 ; GOTO 880
850 IF LEFT $ (R $, 1) = « V » THEN C = 0 ; GOTO 880
860 PRINT « JE N'AI RIEN COMPRIS »
870 GOTO 780
880 R $ = MID $ (R $, 2)
890 IF R $ = « » THEN 860
900 IF ASC (R $) < 48 THEN 880
910 IF ASC (R $) > 57 THEN 880
920 D = ASC (R $) - 48
930 R $ = MID $ (R $, 2)
940 IF R $ = « » THEN 860
950 IF ASC (R $) < 65 THEN 930
960 IF ASC (R $) > 64 + N9 THEN 930
970 E = ASC (R $) - 64
990 REM REGARDONS SI LA TRANSACTION EST POSSIBLE
1 000 F = A (A1, E)
1 010 G = S (A1)
1 020 H = V (1, E)
1 030 IF C = 1 THEN 1230 : REM CAR ACHAT D'ACTIONS
1 040 IF F > = D THEN 1 070
1 050 PRINT « TU N'AS QUE » ; F ; CHR $ (E + 64)
1 060 GO TO 780
1 070 IF H > D THEN 1 100
1 080 PRINT « NON. ELLES NE VALENT PAS ASSEZ »
1 090 GO TO 780
1 100 H = H - D
1 110 F = F - D
1 120 G = G + H * D
1 130 G = INT (G - 1 - D/2)
1 140 IF G > = 0 THEN 1 170
1 150 PRINT « NON. TU N'ES PAS ASSEZ RICHE »
1 160 GO TO 780
1 170 PRINT « O.K. » ;
1 180 IF C = 1 THEN Q (A1, E) = H
1 190 A (A1, E) = F
1 200 S (A1) = G
1 210 V (1, E) = H
1 220 GO TO 780
1 230 IF H + D < 100 THEN 1 260
1 240 PRINT « NON, ELLES VALENT TROP CHER »
1 250 GO TO 780
1 260 IF F + D < 100 THEN 1 290
1 270 PRINT « MAIS TU AS DÉJÀ » ; F ; CHR S (E + 64)
1 280 GO TO 780
1 290 H = H + D
1 300 F = F +D
1 310 G = G - D * H
1 320 GO TO 1 130
1 330 REM RECHERCHE DU JOUEUR SUIVANT
1 340 A1 = A1 + 1
1 350 IF A1 > J9 THEN A1 = 1
1 360 IF A1 <> A THEN 770
1 390 REM AFFICHAGE DES PORTEFEUILLES DE CHACUN
1 400 GO SUB 5 000
1 410 PRINT TAB (10); « PORTEFEUILLES »
1 420 PRINT
1 430 PRINT TAB (6); « NOMBRE ET P.A. »
1 440 FOR I = 1 TO J9
1 450 PRINT TAB (10 * I - 2) ; N $ (I) ;
1 460 NEXT I
1 470 PRINT
1 480 FOR I = 1 TO N9
1 490 PRINT CHR $ (I + 64) ;
1 500 FOR J = 1 TO J9
1 510 IF A (J, I) = 0 THEN 1 540
1 520 PRINT TAB (J * 10 - 2) ; A (J, I) ;
1 530 PRINT TAB (J * 10 + 1) ; Q (J, I) ;
1 540 NEXT J
1 550 PRINT
1 560 NEXT I : PRINT
1 570 PRINT « SOMMES : » ;
1 580 FOR I = 1 TO J9
1 590 PRINT TAB (10) ; N $ (I) ; « : » ;
1 600 PRINT S (I) ; « F »
1 610 NEXT I : PRINT
1 630 REM DÉLAI POUR LECTURE
1 640 FOR I = 1 TO 1 000 : NEXT I
1 660 IF RND > P8 THEN 1 750
1 670 PRINT « DÉVALUATION DE LA MONNAIE »
1 680 FOR I = 1 TO J9
1 690 S (I) = INT (S (I) * 0.8)
1 700 NEXT I
1 710 FOR I = 1 TO 500 : NEXT I
1 730 GO TO 1 400
1 740 REGARDONS SI LA PARTIE EST TERMINÉE
1 750 C = 1
1 760 FOR I = 1 TO J9
1 770 IF S(I) > S (C) THEN C = I
1 780 NEXT I
1 790 IF S(C) > = 5 000 THEN 2 000
1 800 REM PERSONNE N'A GAGNÉ DONC ON CONTINUE
1 810 C = 0 : D = 10 : E = 4
1 820 IF RND < P9 THEN C = 1 : D = 15 : E = 20
1 830 FOR I = 1 TO N9
1 840 FOR I = 1 TO N9
1 840 V (2, I) = INT (RND * D - E)
1 850 IF V (1, I) + V (2, I) < 1 THEN V (2, I) = 1 - V (1, I)
1 860 IF V (1, I) + V (2, I) > 99 THEN V (2, I) = 99 - V (1, I)
1 870 V (1, I) = V (1, I) + V (2, I)
1 880 NEXT I : GO TO 600
1 990 REM LA PARTIE EST TERMINÉE
2 000 GOSUB 5 000
2 010 PRINT « BRAVO POUR LE GAGNANT »
2 020 PRINT N $ (C) ; « EST UN FAMEUX FINANCIER »
2 030 PRINT
```

```
2 040  PRINT « VOICI LES RÉSULTATS »
2 050  PRINT TAB (10); « LIQUIDE ACTIONS TOTAL »
2 060  PRINT
2 070  FOR I = 1 TO J9
2 080  PRINT N $ (I);
2 090  PRINT TAB (12); $ (I);
2 100  C = 0
2 110  FOR J = 1 TO N9
2 120  C = C + V (1, J) * A (I, J)
2 130  NEXT J
2 140  PRINT TAB (18); C;
2 150  PRINT TAB (24); C + $ (I); « F »
2 170  NEXT I : PRINT
2 190  INPUT « ON RECOMMENCE », R $
2 200  IF LEFT $ (R $, 1) = « 0 » THEN 400
2 210  GOSUB 5 000
2 220  PRINT TAB (10); « AU REVOIR »
2 230  PRINT
2 240  STOP
4 990  REM EFFACEMENT DE L'ECRAN
5 000  PRINT CHR $ (16); CHR $ (19); CHR $ (16); CHR $ (22);
5 010  RETURN
9 990  END
```

Le but du jeu est donc de profiter au maximum de ces évolutions des cours pour atteindre le premier la somme plafond fixée à 5 000 francs. Chacun peut suivre la tactique qui lui convient, le programme se bornant à tenir à jour les cours actuels ainsi que les portefeuilles des joueurs. Donc, aucune règle ne permet de gagner à coup sûr ; seuls le bon sens et une certaine habitude peuvent améliorer les performances.

Le réalisme est encore accru par les prélèvements de l'agent de change. Celui-ci n'apparaît pas explicitement pendant la partie, mais s'octroie malgré tout une part du gâteau à chaque transaction : un fixe de 1 F, plus un pourcentage correspondant à 1/2 F par action changeant de main. Ainsi, il est préférable d'éviter des ventes qui rapportent peu.

Quoi qu'il en soit, toutes les valeurs sont paramétrées, et peuvent être modifiées en remplaçant une seule ligne dans le listing en Basic ci-contre.

## Le programme

Le seul sous-programme utilisé se trouve à la ligne 5000. Il est destiné, avec ces valeurs, à changer de page automatiquement sur les terminaux qui disposent de cette facilité, ou d'effacer l'écran pour une présentation plus agréable. Les caractères de contrôle correspondant à ces valeurs sont en général spécifiques à chaque appareil.

Le programme commence donc par un rappel des règles : lignes 100 à 190.

Pour l'affectation des valeurs à plusieurs paramètres, ainsi que la réservation de place pour les tableaux utilisés par la suite : lignes 200 à 320. Chacune des variables comporte son explication.

Les lignes 320 à 360 permettent d'indiquer au programme les noms des joueurs.

La phase d'initialisation du jeu est alors terminée, et les parties successives commencent à la ligne 400 : mise à zéro du nombre d'actions et allocation d'une somme de 2 000 F par joueur.

Puis de la ligne 470 à la ligne 500, affectation aléatoire des cours de départ, dans une fourchette comprise entre 20 et 40 F. La variable C est le drapeau (flag) qui indique la présence éventuelle d'un effondrement des cours.

De la ligne 620 à la ligne 730, affichage de la valeur des actions et de leur variation par rapport aux cours précédents. La ligne 750 tire au sort le joueur qui doit commencer. Ceci afin d'éviter de favoriser toujours la même personne du fait que les prix des actions sont réajustés immédiatement après une opération d'achat ou de vente.

Le programme affiche donc le nom du joueur, puis attend l'indication de la transaction à effectuer ; la réponse attendue est de la forme « JE VENDS 4 D », ou « J'ACHÈTE 3 C ». Les lettres prises en compte sont uniquement A et V. Il est donc possible de répondre plus succinctement : « V 4 D », ou même « A 3 C ». Une réponse nulle (ligne 800) indique la fin des opérations et permet de passer au joueur suivant. Le moindre doute est éliminé par la ligne 860.

Les lignes 1 000 et 1 320 vérifient la validité de l'opération désirée, à savoir :

- On ne peut vendre plus d'actions que l'on en a.
- On ne peut vendre des actions qui ne rapportent rien.
- Il faut disposer de la somme nécessaire pour acheter.
- On ne peut acquérir des actions valant plus de 100 F.
- On ne peut posséder plus de 100 actions du même groupe.

Lorsque tout est contrôlé, le nouveau prix de l'action est calculé, ainsi que la somme restant au joueur après l'opération. L'agent de change prélève sa part à la ligne 1 130.

Les joueurs sélectionnent successivement ce qui les intéresse (lignes 1 330 à 1 360) puis vient la récapitulation des avoirs (lignes 1 400 à 1 640). S'il y a une dévaluation de la monnaie, les nouveaux avoirs sont réaffichés (une fois sur 20 environ).

Le test de fin de partie (lignes 1 750 à 1 790) consiste à comparer l'avoir d'un joueur avec son successeur et à conserver le numéro du plus riche. En fin de boucle, on regarde si le plafond de 5 000 F est alors atteint.

S'il ne l'est pas encore, le jeu continue en affectant une variation aléatoire aux cours. La fourchette est limitée à – 4, + 5 F par tour, sauf dans le cas d'un crash (ligne 1 820) où la fourchette est alors de – 20 à – 6 francs. Le nouveau prix des actions est alors calculé, avec un minimum de 1 F et un maximum de 99 F. Le programme revient à la ligne 600 où ces nouveaux cours sont affichés.

La fin de partie consiste à faire le bilan des sommes en liquide et en actions de chacun des joueurs.

Le programme assez volumineux peut être allégé en supprimant les explications de début de partie, ainsi que tous les commentaires. Aucun branchement n'est effectué à ces lignes.

La durée d'une partie est assez longue (plus d'une heure), surtout si le nombre de joueurs est élevé. Vous pouvez la diminuer en autorisant des variations de cours plus importantes (lignes 1 810 à 1 820). A chacun de profiter au mieux de la situation. ■

**H. EYMARD-DUVERNAY**

# le premier pas chez les microprocesseurs c'est avec notre MAZEL II

Système français pour démarrer votre éducation micro-électronique.

**Liste des matériels disponibles (Fév. 79) :**

| | | | |
|---|---|---|---|
| réf. 50-10 | carte | Micro MAZEL II | - Prix T.T.C. 1990 F |
| – 50-20 | – | Alimentation MAZEL II | - Prix T.T.C. 320 F |
| – 50-40 | – | Vidéo TV MAZEL II | - Prix T.T.C. 1170 F |
| – 50-60 | – | Mémoire C MOS 1K | - Prix T.T.C. 1760 F |

etc...

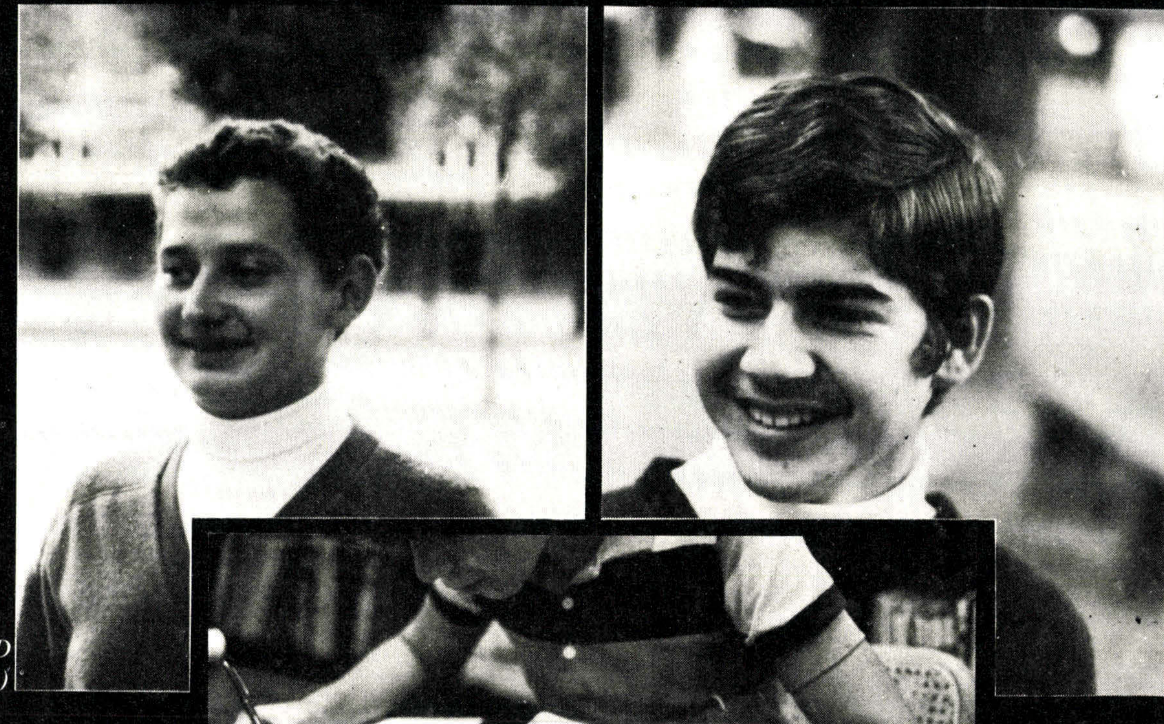






36 RUE DES GRANDS CHAMPS - 75020 PARIS - TÉL. (1) 379.48.51 - TÉLEX 240645 F PAINFOR

# L'informatique, un jeu d'enfant !



*Jean-Marc FRIAUD (15 ans)*

*Sylvain BESSON (16 ans)*

*Eric (15 ans)*

**Système conversationnel pour l'étude du langage LSE**

Le 17 novembre se sont tenues, au Palais de la Découverte, les assises du Prix scientifique Philips pour les jeunes.

Le troisième prix a été remporté dans la discipline Informatique par Jean-Marc Friand et Sylvain Besson, âgés respectivement de 15 et 16 ans et élèves au Lycée Banville de Moulins.

Les deux lauréats ont conçu un programme pour faciliter l'étude du langage symbolique d'enseignement (L.S.E.).

Ils disposaient d'un ordinateur Mitra 15, le programme L.S.E. étant enregistré sur disque magnétique, le but de leur étude étant de connaître la composition de ce programme.

**Un micro-informaticien de quinze ans**

Eric a 15 ans et il fréquente le lycée de Berkeley en Californie. Voici près d'un an, il s'est intéressé aux microprocesseurs et commença à s'y initier avec des camarades au club de son lycée.

Ayant acquis très vite, au sein de celui-ci, les principes de base, il consacra ses loisirs à la réalisation de sa propre carte micro-ordinateur.

Il utilise désormais son système comme carte de base et développe ses cartes mémoires et interfaces.

Alors que ses amis achètent plus volontiers une moto, Eric préfère consacrer les 4 000 F qu'il a péniblement économisés à l'acquisition d'un micro-ordinateur.

**Sont-ils en avance d'une génération ? Non.**
**Si vous redoutez de prendre du retard sur la leur,**
**n'hésitez pas à participer avec Micro-Systèmes au**

# Concours "Micro"

Mission à l'Informatique 24, rue de l'Université, 75007 PARIS

## Micro-Systèmes à Londres

Du 31 janvier au 2 février 1979 se tenait à Londres la deuxième édition de l'exposition « MICROSYSTEMS » qui devait rassembler 60 firmes parmi lesquelles figuraient I.T.T., G.E.C. Elliott, Ferranti, Hewlett Packard...

De par l'impact qu'il impliquera sur la société, le développement de la micro-électronique est souvent qualifié de seconde révolution industrielle. Si les utilisations potentielles des microprocesseurs sont pratiquement illimitées, seul un petit nombre d'ingénieurs semble vraiment informé de toutes les applications possibles, pense-t-on en Angleterre. C'est afin d'alerter les industriels sur la nature de cette révolution et leur montrer les réalisations issues de cette nouvelle technologie que cette exposition s'est tenue.




*Jean-José Wanègue (à gauche) et Alain Tailfiar présentaient, sur le stand de Personnel Computer World, la revue Micro-Systèmes à Londres.*

---

Ce pont entre les fabricants et les utilisateurs de micro-électronique devait offrir simultanément une suite de conférences dont l'ouverture fut marquée par la présence d'un représentant du ministère de l'Industrie venu annoncer les dernières mesures prises par le gouvernement anglais en faveur de l'industrie électronique.

Micro-Systèmes ne pouvait manquer pareille occasion pour porter outre-Manche le témoignage de ce qu'est en France le dynamisme et la compétence du monde de la micro-informatique.

Grâce à M. SOLOMON, rédacteur en chef de « **Personal Computer World** » dont nous avons pu partager le stand, Micro-Systèmes a donné le meilleur exemple de ce que peut être notre presse technique française si l'on en croit l'intérêt qu'elle a suscité chez les quelques milliers de visiteurs de cette exposition.

## Séminaires :

### Cours de Basic

Euro Computer Shop annonce un cours de 2 jours de Basic (Workshop) qui aura lieu les 15 et 16 février, au prix de FF 1 400. Le nombre maximum de participants est fixé à 20. Plusieurs systèmes seront à la disposition des utilisateurs.

Les inscriptions peuvent être prises à :

**Euro Computer Shop**
**16, rue Louis-Pasteur, 92100 Boulogne**
**Tél. : 825.82.52.**

### Cours sur les microprocesseurs

Siemens S.A. organise des journées d'information essentiellement axées sur l'utilisation des microprocesseurs en milieu industriel.

Ces cours auront lieu en 3 sessions, du 18 au 20 avril, du 16 au 18 mai et du 13 au 15 juin 1979, au siège social de Siemens :

39-47, bd Ornano
93200 Saint-Denis

Les deux premières journées de ces sessions sont plus particulièrement destinées à des personnes confrontées aux microprocesseurs au cours de leur vie professionnelle.

La troisième journée sera consacrée entièrement aux possibilités du système modulaire Siemens SMP, construit autour du SAB 8080. Cette journée est également proposée le 22 mars 1979.

Le montant de la participation est de 300 F TTC.

Pour tous renseignements,
**Mlle Roualet**
**Tél. : 820.61.20, poste 2557.**

### L'informatique à l'heure du « libre service » au Palais de la Découverte

C'est à une expérience originale que se sont livrés avec succès les responsables du **Palais de la Découverte** durant les congés scolaires de Noël. Pour la première fois en France il a été offert au public d'utiliser des micro-ordinateurs placés en « libre-service » pour leur plus grand plaisir.

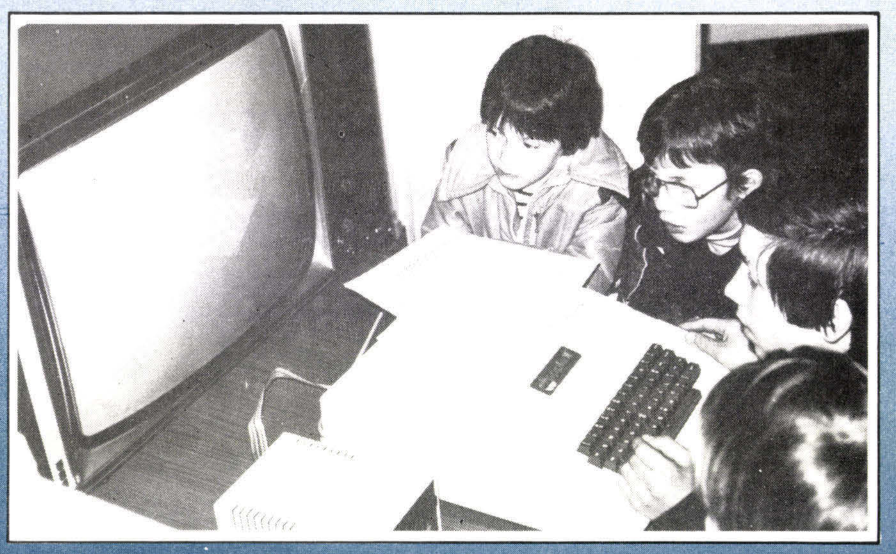
Pour ce faire, les visiteurs trouvaient à leur disposition une large gamme de matériels parmi lesquels on notait la présence d'un **Compucolor,** d'un **Apple II,** de deux **PET,** d'un **Sorcerer** et d'un TRS80 regroupés dans un atelier de Basic. D'autre part une équipe de l'INRP et de l'Institut de Programmation de Paris VI[e] présentait deux systèmes LOGO. L'un servait en construction graphique, l'autre commandait un automate appelé « Tortue ».

Afin de permettre à davantage de jeunes d'accéder à l'informatique grâce à des activités simples et récréatives, ces systèmes LOGO étaient complétés par des programmes de jeux tournant en permanence sur un micro-ordinateur. Ces deux applications répondirent parfaitement à la mission qui leur avait été assignée compte-tenu de la fréquentation assidue d'enfants qui bien souvent avaient tout juste l'âge d'aller à l'école primaire.

Pour les plus âgés, les micro-ordinateurs de l'atelier de Basic étaient à leur disposition sur des périodes qui pouvaient atteindre plusieurs heures, après quoi la plupart repartaient en ayant définitivement tuer le mythe de l'ordinateur. Ils avaient fait la démonstration que l'utilisation d'un Basic en anglais ne pose pas de problème : l'assimilation d'un mini-vocabulaire anglo-saxon s'est faite pour eux avec une rapidité qui a même surpris les organisateurs.

Une telle expérience doit son succès à la présence continue de huit démonstrateurs dont les motivations personnelles pour cette informatique « nouvelle vague » leur out permis d'initier efficacement plus de cinq cents personnes à l'utilisation des ordinateurs. Dans l'immédiat aucune machine n'est encore prête à remplacer l'éducateur, c'est ce que tend à nous prouver cette manifestation.

Après une période de repos forcé de sa section Mathématiques dûe à des travaux de rénovation, le Palais de la Découverte verra le micro-informatique y prendre place tant par la création d'un nouvel atelier, que par l'organisation d'autres expositions ou comme aide à la présentation d'expérimentations dans d'autres salles.



**Microtel annonce dans ses projets la construction du microordinateur « Micro-Systèmes 1 »**

Dans son dernier bulletin de liaison, **Microtel** annonce, outre les différentes activités du club (Cycles de conférences, applications, cours sur le basic, l'électronique...) le projet de construction de « Micro-Systèmes 1 ».

**Arnaud Delmas** (788.76.96) et **J.C. Reynaud** (644.93.18) coordonnent la réalis^tion des projets individuels.

Un exemplaire sera monté dans le local du club. Les commandes de matériels seront groupées (liste des composants déjà disponible) et une première réunion d'information sera prochainement organisée.

Les locaux de Microtel-Club sont situés pour Paris, au 37, rue du Général-Leclerc, 92131 Issy-les-Moulineaux et sont ouverts pour l'instant tous les soirs de 18 heures à 21 heures environ (sauf le vendredi) et le samedi (de 9 h 30 à 12 h 30 et de 14 h 30 à 17 h 30).

Le président est M. Joseph Rinaudo (566.37.38) et tous renseignements peuvent être obtenus au 544.70.23. Concernant la section de Montpellier les renseignements sont à prendre auprès de M. Brabet (16.67 — 63.90.00 poste 358).

**Du nouveau pour le commerce de détail le système de terminaux de magasins IBM 5260**

La Division des Systèmes de Grande Diffusion (DSGD) d'IBM France annonce aujourd'hui un terminal-caisse **autonome** et trois programmes d'applications pour le commerce de détail.

Ce terminal est destiné principalement aux entreprises possédant des points de vente plus ou moins importants et décentralisés.

Ce nouveau système IBM 5260, aux caractéristiques originales, complète la gamme des matériels commercialisés depuis plusieurs années par la DSGD, afin de mettre à la disposition des entreprises une informatique bien adaptée à leurs besoins,

# MICROPROCESSEURS

## J'ai même rencontré l'assistance technique!

*nous a dit un Client!*



Les microprocesseurs, les outils de développement sont pour beaucoup un domaine nouveau, voire inconnu...

Votre entrée dans cette activité nouvelle doit s'effectuer avec sécurité et assurance; c'est pourquoi GEDIS ne se contente pas de vendre des circuits, des systèmes, des outils de développement, mais guide votre choix et assure avec sérieux le suivi technique.

Vous ne comprenez pas quelques détails, un point vous échappe, pourquoi ceci? comment cela?...

**604.81.70 S.A.M.**

Nous avons mis en place, autour de la famille 6800, une panoplie inédite de systèmes, de logiciels et de produits.

Nous organisons, sur les composants et les outils de développement (TDS, EXORciser) des cours appliqués de formation.

**Oui, nous garantissons l'assistance technique réelle et totale à tous nos clients. Clients, mettez-nous à l'épreuve!**

GEDIS — **BOULOGNE (92100)**
53, rue de Paris
Tél. 604.81.70
Télex 270191

GEDIS — **COURBEVOIE (92400)**
165-171, rue J.-P. Timbaud
Tél. 788.50.13

GEDIS — **Neuillé-le-Lierre**
**MONNAIE (37380)**
Tél. (47) 52.96.07

# Pour votre 6800 :

**MINIPROM I :** programmateur d'EPROM 2708/16. Version "J" : spécialement adaptée au MEK-D2 ...................... 680 F
Version "A" : moniteur indépendant, géré depuis une console (ACIA) ......... 725 F
Logiciels de gestion fournis sur EPROM 2708.
MINIPROM I peut aussi programmer des 2716 grâce aux programmes de gestion :
"J - 2716" ...................... 125 F
"A - 2716" ...................... 180 F
fournis sur EPROM 2708.
MINIPROM I s'interface facilement avec les autres ordinateurs de la gamme 6800 MOTOROLA par PIA. Nous contacter.

**Cartes MOTOROLA**

| | |
|---|---|
| 16K EPROM vide | 1450 F |
| EPROM 2708 | 85 F |
| Les 8 | 615 F |

**Lampe UV :**
efface 4 EPROM simultanément ..... 555 F

**Logiciels 6800 "COMPUTER WORKSHOP"**

| | |
|---|---|
| • Editeur Assembleur<br>• BASIC 8 K<br>• Relocalisateur<br>• Jeux, etc. | Compatibles "MIKBUG". Cassettes ou ruban perforé. |

**Initiation MOTOROLA :**

| | |
|---|---|
| Kit MEK-6800-D2 | 1530 F |
| ou plus : MEK-D2 + supports CI + 256 oct + RAM + buffers | 1850 F |
| Carte 16K RAM + kit de "Handshake" | 2820 F |
| Panier 5 cartes avec connecteurs | 995 F |
| Minibus à 2 connecteurs | 295 F |

Alimentations 3 tensions (5 V, 12 V, -12 V) pour micro-ordinateur (prix selon puissance).

| | |
|---|---|
| MC 6800 P | 102 F |
| MC 6802 P | 153 F |
| MCM 68 A 10 P (450 ns) | 30 F |
| MC 6820 P | 45 F |
| MC 6850 P | 37 F |
| MC 6840 P (Timer) | 145 F |
| MC 6844 L (DMA) | 187 F |
| MC 6875 L (horloge) | 67 F |
| RS 232 1488 | 17 F |
| RS 232 1489 | 17 F |
| SFF 96364 (CRT) | 162 F |
| 4 N 33 | 17 F |
| Convertisseur D/A, 8 bits, 300 ns MC 1408 L8 | 43 F |
| RAM 1 k x 4 (C 2142) | 100 F |
| RAM 16 k x 1 dynamique 4116 P 3 | 155 F |

Tous nos prix s'entendent hors taxes (T.V.A. 17,60 %) - Règlement à la commande + 15 F de frais de port et d'emballage.

**MPU** est distribué par : **SELFCO,** 31, rue du Fossé des Treize 67000 Strasbourg.




**Documentation sur simple demande**
**12, rue Chabanais - 75002 Paris - tél. 261.81.03.**

simple et d'un coût intéressant.

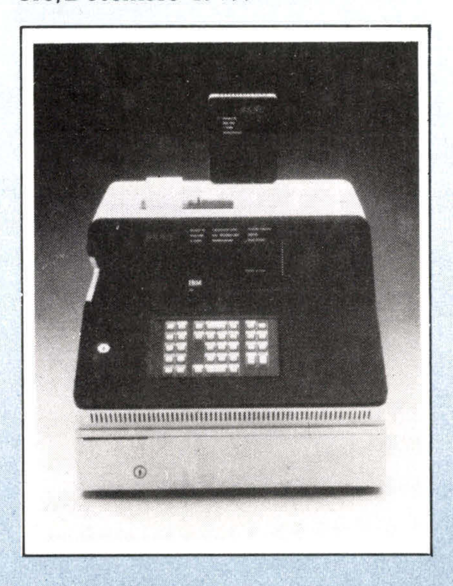
Ce système de terminaux de magasin existe en trois versions :

**Le terminal autonome IBM 5265** comprend : un clavier, un écran d'affichage, deux stations d'impression, un guide-opérateur, une unité de calcul et de contrôle et un minidisque.

**Le terminal principal IBM 5265** a les mêmes caractéristiques que le précédent, avec en plus un dispositif qui permet de lui connecter des terminaux satellites.

**Le terminal satellite IBM 5266** a les mêmes caractéristiques que le 5265 à l'exception du mini-disque.

Le système IBM 5260 sera fabriqué à l'usine de Vimercate (Italie). Les premières livraisons en clientèle sont prévues pour Novembre/Décembre 1979.



## Un compilateur Fortran pour les systèmes universels de développement 8002 de Tektronix

La Société Microsoft propose un compilateur « Fortran 80 » aux utilisateurs du système de développement 8002 de Tektronix.

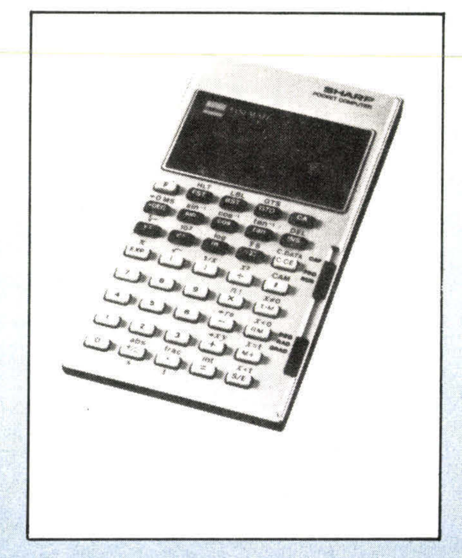
Ce compilateur, écrit pour microprocesseurs 8080, 8085 et Z80, est comparable à celui que l'on trouve pour des mini-ordinateurs. Il facilite le développement de programmes nécessitant des calculs arithmétiques ; outre les fonctions standard telles que les fonctions trigonométriques ou exponentielles, ce compila-

teur Fortran dispose de l'addition, soustraction, multiplication, division, etc. sur 32 ou 64 bits en virgule flottante. Ces sous-programmes sont parmi les plus rapides pour l'exécution sur 8080.

**Tektronix**
**Z.I. de Courtabœuf**
**Avenue du Canada, B.P. 13, 91401 Orsay cedex**
**Tél. : 907.78.27.**

## Calculatrice programmable de poche

Référencée PC 1201, cette calculatrice programmable, conçue par Sharp, peut être programmée jusqu'à 128 pas grâce à un dispositif d'instruction par ordre au lieu du système standard 1 touche/1 pas, de façon à ce que plusieurs pas puissent être couverts par une commande unique.



D'une capacité illimitée, avec 12 mémoires indépendantes, fonctionnant même si l'alimentation est coupée ; 37 fonctions incorporées pour des calculs scientifiques, mathématiques et statistiques instantanées, les programmes peuvent être divisés en 13 groupes, comprenant une fonction de sous-programme.

Chaque introduction de calcul est confirmée par un « Bip » sonore et la lecture s'effectue sur 10 chiffres.

**Sharp**
**151-153, avenue Jean-Jaurès, 93300 Aubervilliers.**
**Tél. : 834.93.44.**



OK. MACHINE
and TOOL CORP. BRONX NY
(U.S.A.)

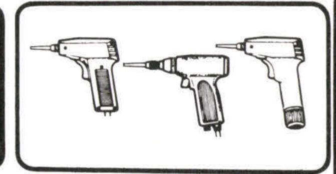
WRAPPING
À L'ÉCHELLE
INDUSTRIELLE

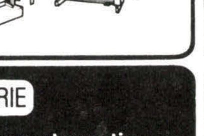


INDUSTRIE
Outils à main



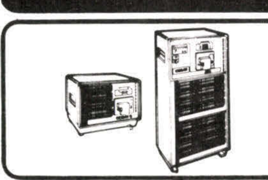
INDUSTRIE
Pistolets
mécaniques
électriques
pneumatiques

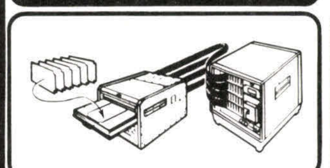




INDUSTRIE
Machines
semi-automatiques

INDUSTRIE
Machines automatiques
de contrôle
de production



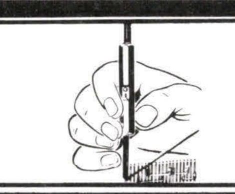


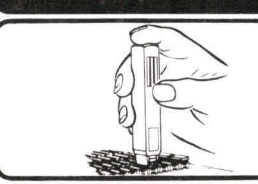
INDUSTRIE
Cadres pour
prise de lecture

TECHNIQUE
WRAPPING

SERVICE
LABORATOIRE

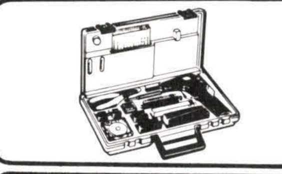
LABORATOIRE
Outil à main*
combiné
3 opérations

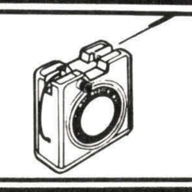




LABORATOIRE
Outils à insérer
les C.I.

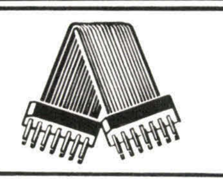
LABORATOIRE
Ensembles
outillage
et fournitures





LABORATOIRE
Distributeurs de fil*
coupe-dénudage

LABORATOIRE
Câbles plats avec
supports enfichables
Supports à wrapper



Dans la
qualité
SOAMET
une gamme
complète
de produits
et de
services

*Brevets demandés dans les principaux pays industriels.

Importateur Exclusif **TOUT L'OUTILLAGE POUR L'ELECTRONIQUE**

**SOAMET s.a.** 10, Bd. de la Mairie - 78290 CROISSY-s/SEINE - 976.55.72 976.24.37

GIROTYPO - BAGNEUX - 665-38-23

## Deux nouvelles mémoires programmables en technologie Schottky

Texas Instruments vient d'ajouter à la gamme de ses mémoires bipolaires, deux nouvelles PROM de 8192 bits. Désignées par les références SN 74 S 478 et SN 74 S 479, les nouvelles mémoires programmables sont organisées en 1024 mots de 8 bits.

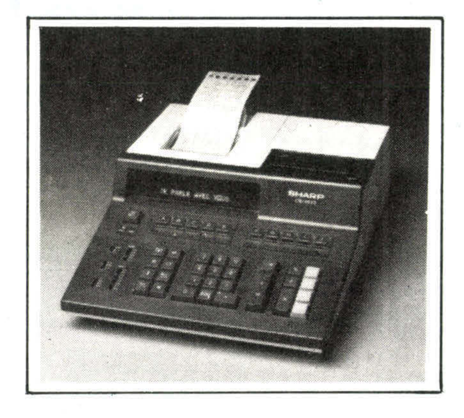
Parmi leurs principales caractéristiques, on peut citer un temps d'accès typique de 45 ns, un temps d'accès maximum de 70 ns, un temps de validation maximum de 40 ns, et une consommation de 630 mW en typique, soit environ la moitié de la puissance consommée par 2 PROM de 4 K bits. Les nouvelles mémoires disposent de lignes multiples de sélection afin de simplifier l'expansion mémoire ; les fusibles de programmation sont réalisés en titane-tungstène une technologie déjà bien éprouvée ; les entrées mémoires sont du type PNP donc à faible courant ; deux versions disponibles pour les sorties : collecteur ouvert (modèle SN 74S 479) où sorties en logique trois états (modèle SN 74S 478).

L'organisation en 1024 mots de 8 bits rend ces nouvelles mémoires particulièrement intéressantes pour les systèmes à base de microprocesseurs et pour les applications de mémoires de contrôle à microprogrammation où le nombre de boîtiers doit être réduit.

Disponible en boîtier standard à 24 broches, les nouvelles PROM sont interchangeables avec les mémoires programmables de 4K bits de Texas Instruments SN 74S 474 et SN 74S 475. Elles sont présentées en boîtier plastique (suffixe N) ou céramique (suffixe J).

## Réalisation de la paie accessible à tous

Par son écran alpha numérique, le système 4520 de Sharp converse avec l'opérateur. Son programme se déroule et indique au fur et à mesure à l'utilisateur la marche à suivre rubrique après rubrique. D'autre part, sa capacité programme, 4096




instructions par cassette interchangeable, lui permet d'automatiser toutes les formes de paies existant en France.

Enfin sa capacité de 117 mémoires autorise la conservation de toutes les constantes nécessaires, la totalisation de l'ensemble des rubriques de la paie, éventuellement ventilées pour la comptabilité analytique.

## Programmeur de PROM

Electronic J.L. vient de se voir confier la représentation exclusive de la société International Microsystems Inc spécialisée dans la programmation de mémoire PROM.

Deux systèmes sont disponibles :

• **Un programmateur universel** de mémoires PROM référencé **IM 1000.** Contrôlé par microprocesseur il permet la programmation par son clavier hexadécimal, par connection sur un calculateur, par instructions en provenance de télétype.

Visualisation par 14 chiffres des fonctions, des adresses et des données présentée sur afficheurs à gaz.

Il réalise les fonctions : lecteur, programmation, copie, vérification et télécommande.

Il programme par modules personnalisés les mémoires des principaux constructeurs : AMD, Electronics Arrays, Fairchild, Intel, Intersil, Mostek, Motorola, NMI, NS, Raythéon, Signetics, Texas, etc.

• **Un « Gangprom programmer »** qui permet la programmation de 16 PROM simultanément. Modèle IM 2020.

**Electronic J.L.**
**97, rue des Chantiers, 78000 Versailles**
**Tél. : 950.28.20.**

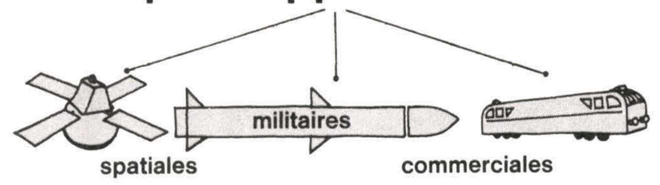
## Index des Annonceurs





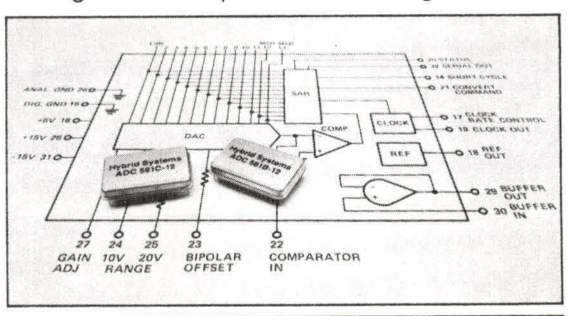
# Hybrid Systems
CORPORATION

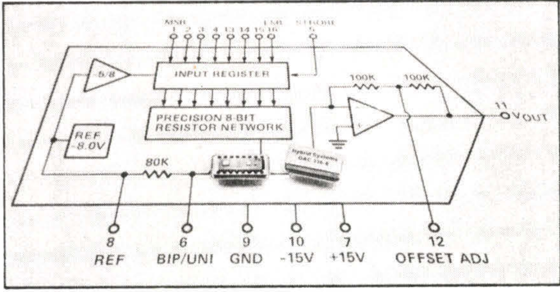
## CONVERTISSEURS
## N/A et A/N
### pour applications




## ADC 581
seconde source ADC-85

12 bits - boîtier métallique 32 broches DIL
faible consommation - 750 mW seulement
sortie série ou parallèle
buffer incorporé
horloge interne - possibilité horloge externe




birépub 638

## DAC 336
DAC 8 et 12 bits entièrement complet
absolument aucun réglage
registre tampon incorporé
compatible Bus µP
linéarité 1/8 LSB
CMOS ou TTL - sortie tension programmable




16, rue Baudin, 92300 LEVALLOIS
Tél. : 758.59.10 - Télex 630 065

**Venez nous voir au salon des composants : pavillon américain**

Salon des composants : Allée 14 - Stand 120

# MK 14 KIT MICRO PROCESSEUR SC/MP
## distribué par JCS composants



Compte tenu du succès de cet appareil, un certain délai de livraison peut être nécessaire.

**795,00 F. TTC**
**676,00 F. HT**

### UN PRIX JAMAIS ATTEINT

Pour moins de 800 F, ce microprocesseur en KIT place la micro-informatique à la portée de tous les hobbyistes, les étudiants, les techniciens.

**CARTE DE BASE**

- Microprocesseur SC/MP
- Clavier hexadécimal
- Bloc afficheur 8 digits
- Moniteur 512 octets
- Supports C.I. MOS
- RAM 256 octets
- Horloge 4 MHz
- 16 E/S parallèles
- Régulateur 5 V.
- Circuit époxy

**MANUEL EN FRANÇAIS**

Le manuel de montage et de programmation livré avec l'appareil est en français. Il donne plus de 80 pages d'explications détaillées de montage et de fonctionnement. Le MK 14 est immédiatement utilisable grâce aux programmes fournis dans différents domaines tels que jeux, musique, calcul, électronique...

**OPTIONS**

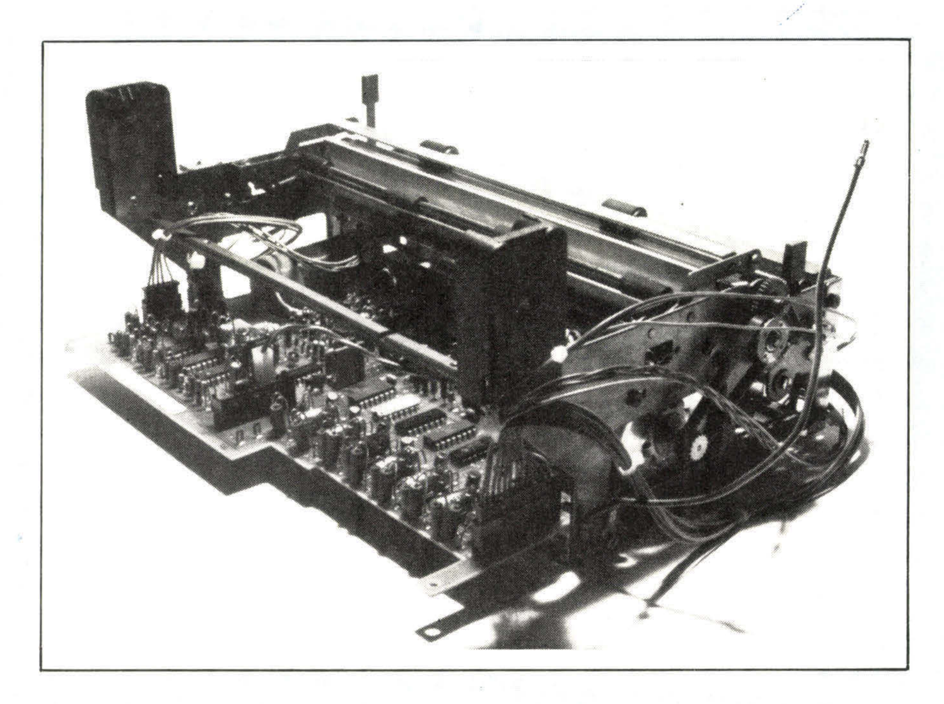
- MEMOIRE : par simple mise en place sur la carte de 3 RAM supplémentaires, 384 octets s'ajoutent à la version de base .......... 198,00 F
- INTERFACE CASSETTE : elle permet le stockage et la lecture sur mini-cassette des programmes élaborés par l'utilisateur .......... 120,00 F
- SUPER-MONITEUR : version améliorée du moniteur de base, il facilite la lecture, l'écriture sur cassette, permet l'exécution des programmes pas à pas, rend plus aisée l'entrée des programmes en mémoire .......... 145,00 F

Liste des revendeurs

**CHAMPREMIER** 42300 ROANNE
**CHARLET** 47000 AGEN
**C.S.E.** 57000 METZ
**DECOCK** 59000 LILLE
**ELECTROME** 33000 BORDEAUX
**EQUIPT. ELEC.** 68100 MULHOUSE
**FANATRONIC** 75015 PARIS
**FANATRONIC** 92000 NANTERRE
**REBOUL** 25000 BESANÇON
**SELECTRONIC** 59000 LILLE
**SELFCO** 67000 STRASBOURG

Epiac : 357.03.30

**IMPORTATEUR :**
**JCS COMPOSANTS**
35, rue de la Croix-Nivert 75015 PARIS Tél. 306.93.69



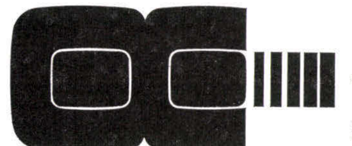
### Mécanisme d'imprimante thermique 80 caractères par seconde

Dataproducts introduit sur le marché un mécanisme d'imprimante thermique à impression matricielle par points. Destiné au marché OEM, ce mécanisme comprend la tête d'impression ainsi que toute la mécanique et l'électronique d'asservissement. Sa mise en œuvre nécessite seulement une alimentation externe de la logique TTL, et un générateur de caractères.

Ce nouveau mécanisme d'imprimante est référencé T-80M. Il s'agit d'une version sans habillage de l'imprimante thermique T-80 silencieuse et sans maintenance, déjà commercialisée.

Ses caractéristiques sont les suivantes :

- Vitesse de 80 caractères par seconde.



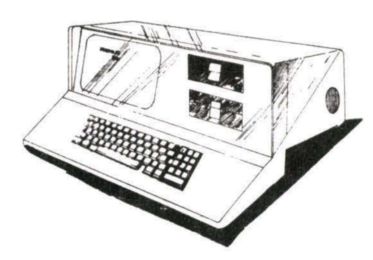
SOCIÉTÉ OCCITANE D'ÉLECTRONIQUE
119, chemin Basso-Cambo
31300 TOULOUSE

Télex 530094 OCCITEL
Tél. : (61) 40-05-15

***NOUS CONCEVONS ET FABRIQUONS POUR VOUS DES SYSTÈMES A BASE DE 6800***

**SYSTÈME X 1**

Un vrai système d'informatique



- ✱ Unité Centrale 6800
- ✱ Vidéo —30 cm avec contrôleur 24 × 80
- ✱ Deux mini-disquettes double densité avec contrôleur
- ✱ Clavier 75 touches
- ✱ 8 K PROM
- ✱ 16 K RAM dynamiques (Ext. : 64 K)

**COMPLET** : 18 800 F.H.T.

**IMPRIMANTE SX1**

- ✱ 110 caractères/sec
- ✱ papier normal
- ✱ disponible mars

**LOGICIEL**

- ✱ LDOS Gestion des disques Création de fichiers
- ✱ LEDIT Éditeur S.O.E.
- ✱ LASS Assembleur S.O.E.
- ✱ LBASIC étendu avec gestion de fichiers sur disquette
- ✱ Logiciels d'applications spécialisées.

**POUR UNE MEILLEURE PROTECTION ET UNE PLUS GRANDE FACILITÉ D'UTILISATION NOUS POUVONS METTRE VOTRE LOGICIEL SUR PROM.**

**CARTES MICROSYSTÈMES**

BUS compatible MOTOROLA

- ✱ **MP1** : Unité centrale 6800 - Interfaces série et parallèle - Sortie RS 232 C - boucle de courant en option.
- ✱ **FSF1** : Carte contrôleur mini-floppy - double densité - drives.
- ✱ **S8-1** : RAM statique 8K - Alimentation + 5, sélection par roue codeuse.
- ✱ **D16-1** : RAM dynamique 16K - Alimentation + 5, — 12, + 12 - sélection par roue codeuse.
- ✱ **R16-1** : REPROM — 16K - Alimentation + 5, — 12, + 12.
- ✱ **R8-1** : REPROM - 8K - Alimentation + 5, — 12, + 12.

**MONITEUR VIDÉO**

30 cm - châssis - tube anti-reflet - vert en option - Alimentation 24 V alternatif.

**BLOC MINI-DISQUETTE**

Double densité avec ou sans contrôleur.

**BLOC ALIMENTATION**

- 5 V, 15 A, + 12 V, 7 A, - — 12 V, 2 A.
- sortie 24 V alternatif - Avec ventilations.

**CLAVIER 75 TOUCHES**

- AZERTY - 52 touches alphanumériques - 12 touches fonction - 11 touches spécifiques.

***NOUS CHERCHONS A COMPLÉTER NOTRE RÉSEAU DE DISTRIBUTEURS***

# système disques souples

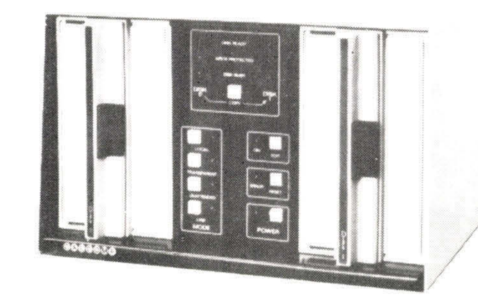
**EIA RS 232 C ou CCITT V 24.**

CARACTÉRISTIQUES :

- 1 ou 2 unités de disques jusqu'à 1,2 millions de caractères en ligne.
- 2 entrées RS 232 asynchrones ASCII.
- Sélection de 110 à 19200 bauds.
- Copie des disquettes.

APPLICATIONS :

- Réduction des coûts Time-Sharing (stockage et édition off-line).
- Mémoire disque pour calculateurs.
- Mémoire de masse pour terminaux.
- Formatage et édition de textes.
- Mémoire de masse connectable sur les systèmes Apple, PET...



**Véritable mémoire de masse CALCOMP 7000**

Autres modèles de floppy : 143 M double face simple ou double densité.

Technitron distribue d'autres terminaux et lancera au "Printemps Informatique" (20 au 23 mars), le micro-ordinateur **DMS Calcomp.**




8, av. Aristide Briand - 92220 Bagneux
Tél. 657.11.47 - Télex 240792

• Tête imprimante constituée d'éléments chauffants en couche épaisse déposée sur un substrat céramique.
• Impression de caractères par matrice de 5 x 7 points, ou impression graphique avec une résolution horizontale et verticale de 70 points par pouce.
• L'imprimante utilise des rouleaux de papier thermo-sensible standard de 8,75 pouces de large.

Pour toute information complémentaire :
**Interco**
**Tél. : 734.73.35 – 566.67.36.**

### Une carte d'interface cassette-audio

Sous la référence 6954-ACI Intersil présente une nouvelle carte d'interface pour l'Intercept Junior.

Ce module permet de connecter directement tout magnétophone à cassette « grand public ». Il fournit ainsi un moyen de stockage de masse peu coûteux au système d'initiation aux microprocesseurs Intercept Junior.

L'utilisation conjointe de la modulation en fréquence (contrôlée par quartz) des informations, des techniques à bouclage de phase et la génération de signaux de niveau élevé permet de stocker de façon sûre 200 000 caractères (par exemple) sur des cassettes audio à bon marché.

Cette carte est disponible chez Tekelec au prix de 1 188 F HT.
**Tekelec**
**Cité des Bruyères, rue Carles Vernet, 92310 Sèvres**
**Tél. : 027.75.35.**

### Logiciel de comptabilité

C.G.I.A. annonce la sortie de la première partie de son logiciel de comptabilité sur micro-ordinateur Alpha Micro « AMS 100 ».

Ce logiciel comporte :

Un système de saisie interactif, avec auto-contrôle des données, gestion dynamique du plan comptable et possibilité de correction des informations saisies à tout instant et à tous les niveaux.

Un système d'édition pour les journaux, le grand livre et la balance.

L'intérêt de ce produit réside dans sa simplicité d'utilisation et dans sa capacité de stockage d'information pratiquement illimité ; en effet, sa mise en œuvre ne nécessite l'utilisation que de deux disquettes (3740) en ligne.

Cette application est en démonstration chez :
**Euro Computer Shop**
**16, rue Louis-Pasteur, 92100 Boulogne.**

# ENFIN

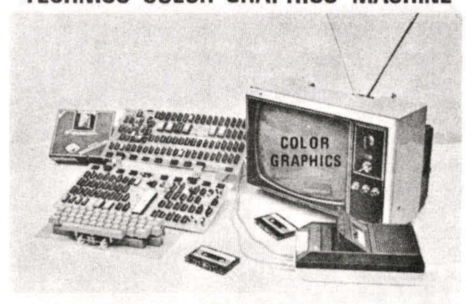
un

micro-ordinateur

## 16 bits

**SUPER SYSTEM 16**

industriel
et scientifique




TMS 9900




- ☐ entrées/sorties RS 232, 32 bits E/S, extension possible jusque 6 RS 232.
- ☐ entrées/sorties parallèles 192 bits E/S.
- ☐ interface Dual Floppy Disk.
- ☐ interface lecteur de cassettes.
- ☐ interface visualisation graphique et alpha-numérique.
- ☐ capacité mémoire 65 K octets, adressable directement.
- ☐ éditeur, assembleur, éditeur de liens, DOS, Basic, Super Basic, Fortran IV.
- ☐ répertoire de 69 instructions.

*Pour tous renseignements :*




**Techinnova 2000**
277, rue Saint-Honoré
**75008 PARIS**
Tél. : 296-35-04

Pour plus de détails, utiliser nos cartes-réponses.




**une réalité grâce
au microprocesseur**



**R 6500 ROCKWELL**

# SYSTEM-CONTACT

- 4, rue des Sœurs - 67810 HOLTZHEIM
  Tél. : (88) 78.20.89 - Télex 890.266 Sycon
- 1, place de la Balance - Silic 473
  94613 RUNGIS CEDEX - Tél. : (1) 687.12.58
  Télex 202.312 Rocsyst

# SYSTEME ANALOGIQUE E/S POUR MICROCALCULATEUR

## EN ACQUISITION DE DONNEES LES CARTES

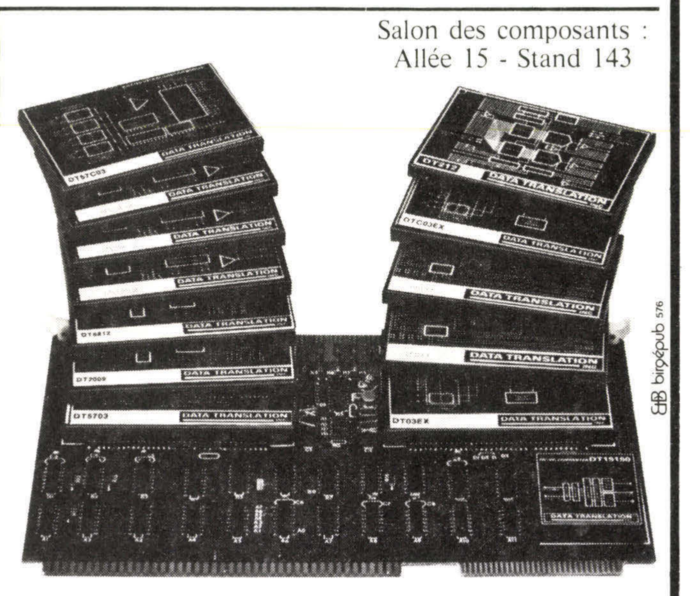
## DATA TRANSLATION INC

## FONT PLUS VITE ET MIEUX

Pour applications industrielles, scientifiques, laboratoires
cartes d'interfaces standard
pour la plupart des µ calculateurs DEC, INTEL, ZILOG, COMPUTER-AUTOMATION et NS.

***PLUS DE PROBLEME DE LOGICIEL !***
***PLUS DE TEMPS D'ETUDE !***

Salon des composants :
Allée 15 - Stand 143




**DATA TRANSLATION**

propose
plus de 20 modules standard d'acquisition de données
et réalise plus de 50 cartes E/S

Entrées isolées (250 V), rapides (125 KHz)
Totale compatibilité 6812/6912

**Prix à partir de 1800 F.h.t.**

Nouveau catalogue 70 pages à :

SACASA

sacasa
société applications composants actifs et systèmes associés

**7, rue de l'Avenir · 92360 MEUDON LA FORÊT**
**Tél.: 630.68.39 · Télex 698 958 F code 209**

# en direct des U.S.A.

## documentation et programmes pour votre **micro ordinateur**

* **Programmes** sur cassettes pour TRS 80, Apple II, PET, échecs, bridge, othello, startrek, etc.
* Fortran pour TRS 80, assembleur, traitement de texte, gestion, etc.
* **Livres** (en anglais) facile à lire Basic pour débutant, initiation aux micros, liste de programmes de jeux, bibliothèque de programme BASIC, etc.
* **Abonnement** aux revues américaines Byte, Kilobaud...
* **Cassettes** C 10 (5 minutes par face) pour micro.

**SIVEA s.a.**
20, rue de Léningrad - 75008 Paris
Tél. 387.59.36

**BON** à remplir et à envoyer à SIVEA MC 1
pour recevoir une documentation "MICRO"

Nom (majuscules) ______ Profession ______
Prénom ______
Adresse complète ______
Code postal ______ Ville ______

# vous aimeriez,

## que l'on enregistre en votre absence

- Un film à ne pas manquer
- Un match avec les Verts
- Un concert exceptionnel
- Une pièce de théâtre
- Une compétition de ski...

**VidéO ACTUALITÉ vous fait passer du rêve à la réalité...**

VIDEO




FAIRCHILD